DOCUMENT RESUME

ED 079 980 EM 011 347

| | |
|---|---|
| AUTHOR | Kimball, Ralph B. |
| TITLE | Self-Optimizing Computer-Assisted Tutoring: Theory and Practice. Technical Report Number 206. |
| INSTITUTION | Stanford Univ., Calif. Inst. for Mathematical Studies in Social Science. |
| SPONS AGENCY | National Science Foundation, Washington, D.C. |
| REPORT NO | IMSSS-TR-206 |
| PUB DATE | 25 Jun 73 |
| NOTE | 141p. |
| EDRS PRICE | MF-$0.65 HC-$6.58 |
| DESCRIPTORS | *Calculus; *Computer Assisted Instruction; Experiments; Higher Education; Learning Theories; *Problem Solving; Research; Technical Reports; *Tutorial Programs; *Tutoring |
| IDENTIFIERS | Heuristics |

ABSTRACT

Research constructed a computer-assisted instruction (CAI) tutor which could transmit problem solving heuristics, choose examples, handle examples from a range of students, and learn superior student heuristics. Using a student subject model and tutorial strategy, an experiment was conducted with 284 problems. Subject response data indicated the information updating and student learning models could be considered separately since response discontinuities were related to instances of entering failure states. Students made 0.101 fewer additional steps per problem worked than the tutor and 0.011 fewer failures per problem step. Students improved 25% of the solutions and the tutor acquired some novel solutions. The research clarified the definition of a tutor in CAI, established a methodology for problem solving heuristics, defined and supported a model of how student heuristics change after failure, implemented a scheme for tutor improvement, and combined the results of research in symbolic integration and algebraic simplification for use in CAI. Future research should investigate the role of failure in strategy changes, convergence in problem solving, quantitative measures in CAI tutoring, and tutors for other subjects. (LB)

FILMED FROM BEST AVAILABLE COPY

ED 079980

# SELF-OPTIMIZING COMPUTER-ASSISTED TUTORING: THEORY AND PRACTICE

EM

BY

RALPH B. KIMBALL

TECHNICAL REPORT NO. 206

JUNE 25, 1973

PSYCHOLOGY AND EDUCATION SERIES

INSTITUTE FOR MATHEMATICAL STUDIES IN THE SOCIAL SCIENCES
STANFORD UNIVERSITY
STANFORD, CALIFORNIA

EM 011 347

FILMED FROM BEST AVAILABLE COPY

## TECHNICAL REPORTS

### PSYCHOLOGY SERIES

### INSTITUTE FOR MATHEMATICAL STUDIES IN THE SOCIAL SCIENCES

(Place of publication shown in parentheses; if published title is different from title of Technical Report, this is also shown in parentheses.)

(For reports no. 1 - 44, see Technical Report no. 125.)

50 R. C. Atkinson and R. C. Calfee. Mathematical learning theory. January 2, 1963. (In B. B. Wolman (Ed.), Scientific Psychology. New York: Basic Books, Inc., 1965. Pp. 254-275)

51 P. Suppes, E. Crothers, and R. Weir. Application of mathematical learning theory and linguistic analysis to vowel phoneme matching in Russian words. December 28, 1962.

52 R. C. Atkinson, R. Calfee, G. Sommer, W. Jeffrey and R. Shoemaker. A test of three models for stimulus compounding with children. January 29, 1963. (J. exp. Psychol., 1964, 67, 52-58)

53 E. Crothers. General Markov models for learning with inter-trial forgetting. April 8, 1963.

54 J. L. Myers and R. C. Atkinson. Choice behavior and reward structure. May 24, 1963. (Journal math. Psychol., 1964, 1, 170-203)

55 R. E. Robinson. A set-theoretical approach to empirical meaningfulness of measurement statements. June 10, 1963.

56 E. Crothers, R. Weir and P. Palmer. The role of transcription in the learning of the orthographic representations of Russian sounds. June 17, 1963.

57 P. Suppes. Problems of optimization in learning a list of simple items. July 22, 1963. (In Maynard W. Shelly, II and Glenn L. Bryan (Eds.), Human Judgments and Optimality. New York: Wiley. 1964. Pp. 116-126)

58 R. C. Atkinson and E. J. Crothers. Theoretical note: all-or-none learning and intertrial forgetting. July 24, 1963.

59 R. C. Calfee. Long-term behavior of rats under probabilistic reinforcement schedules. October 1, 1963.

60 R. C. Atkinson and E. J. Crothers. Tests of acquisition and retention, axioms for paired-associate learning. October 25, 1963. (A comparison of paired-associate learning models having different acquisition and retention axioms, J. math. Psychol., 1964, 1, 285-315)

61 W. J. McGill and J. Gibbon. The general-gamma distribution and reaction times. November 20, 1963. (J. math. Psychol., 1965, 2, 1-18)

62 M. F. Norman. Incremental learning on random trials. December 9, 1963. (J. math. Psychol., 1964, 1, 336-351)

63 P. Suppes. The development of mathematical concepts in children. February 25, 1964. (On the behavioral foundations of mathematical concepts. Monographs of the Society for Research in Child Development, 1965, 30, 60-96)

64 P. Suppes. Mathematical concept formation in children. April 10, 1964. (Amer. Psychologist, 1966, 21, 139-150)

65 R. C. Calfee, R. C. Atkinson, and T. Shelton, Jr. Mathematical models for verbal learning. August 21, 1964. (In N. Wiener and J. P. Schoda (Eds.), Cybernetics of the Nervous System: Progress in Brain Research. Amsterdam, The Netherlands: Elsevier Publishing Co., 1965. Pp. 333-349)

66 L. Keller, M. Cole, C. J. Burke, and W. K. Estes. Paired associate learning with differential rewards. August 20, 1964. (Reward and information values of trial outcomes in paired associate learning. (Psychol. Monogr., 1965, 79, 1-21)

67 M. F. Norman. A probabilistic model for free-responding. December 14, 1964.

68 W. K. Estes and H. A. Taylor. Visual detection in relation to display size and redundancy of critical elements. January 25, 1965, Revised 7-1-65. (Perception and Psychophysics, 1966, 1, 9-16)

69 P. Suppes and J. Donio. Foundations of stimulus-sampling theory for continuous-time processes. February 9, 1965. (J. math. Psychol., 1967, 4, 202-225)

70 R. C. Atkinson and R. A. Kinchla. A learning model for forced-choice detection experiments. February 10, 1965. (Br. J. math stat. Psychol., 1965, 18, 184-206)

71 E. J. Crothers. Presentation orders for items from different categories. March 10, 1965.

72 P. Suppes, G. Groen, and M. Schlag-Rey. Some models for response latency in paired-associates learning. May 5, 1965. (J. math. Psychol., 1966, 3, 99-128)

73 M. V. Levine. The generalization function in the probability learning experiment. June 3, 1965.

74 D. Hansen and T. S. Rodgers. An exploration of psycholinguistic units in initial reading. July 6, 1965.

75 B. C. Arnold. A correlated urn-scheme for a continuum of responses. July 20, 1965.

76 C. Izawa and W. K. Estes. Reinforcement-test sequences in paired-associate learning. August 1, 1965. (Psychol. Reports, 1966, 18, 879-919)

77 S. L. Blehart. Pattern discrimination learning with Rhesus monkeys. September 1, 1965. (Psychol. Reports, 1966, 19, 311-324)

78 J. L. Phillips and R. C. Atkinson. The effects of display size on short-term memory. August 31, 1965.

79 R. C. Atkinson and R. M. Shiffrin. Mathematical models for memory and learning. September 20, 1965.

80 P. Suppes. The psychological foundations of mathematics. October 25, 1965. (Colloques Internationaux du Centre National de la Recherche Scientifique. Editions du Centre National de la Recherche Scientifique. Paris: 1967. Pp. 213-242)

81 P. Suppes. Computer-assisted instruction in the schools: potentialities, problems, prospects. October 29, 1965.

82 R. A. Kinchla, J. Townsend, J. Yellott, Jr., and R. C. Atkinson. Influence of correlated visual cues on auditory signal detection. November 2, 1965. (Perception and Psychophysics, 1966, 1, 67-73)

83 P. Suppes, M. Jerman, and G. Groen. Arithmetic drills and review on a computer-based teletype. November 5, 1965. (Arithmetic Teacher, April 1966, 303-309.

84 P. Suppes and L. Hyman. Concept learning with non-verbal geometrical stimuli. November 15, 1968.

85 P. Holland. A variation on the minimum chi-square test. (J. math. Psychol., 1967, 3, 377-413).

86 P. Suppes. Accelerated program in elementary-school mathematics -- the second year. November 22, 1965. (Psychology in the Schools, 1966, 3, 294-307)

87 P. Lorenzen and F. Binford. Logic as a dialogical game. November 29, 1965.

88 L. Keller, W. J. Thomson, J. R. Tweedy, and R. C. Atkinson. The effects of reinforcement interval on the acquisition of paired-associate responses. December 10, 1965. (J. exp. Psychol., 1967, 73, 268-277)

89 J. I. Yellott, Jr. Some effects on noncontingent success in human probability learning. December 15, 1965.

90 P. Suppes and G. Groen. Some counting models for first-grade performance data on simple addition facts. January 14, 1966. (In J. M. Scandura (Ed.), Research in Mathematics Education. Washington, D. C.: NCTM, 1967. Pp. 35-43.

91 P. Suppes. Information processing and choice behavior. January 31, 1966.

92 G. Groen and R. C. Atkinson. Models for optimizing the learning process. February 11, 1966. (Psychol. Bulletin, 1966, 66, 309-320)

93 R. C. Atkinson and D. Hansen. Computer-assisted instruction in initial reading: Stanford project. March 17, 1966. (Reading Research Quarterly, 1966, 2, 5-25)

94 P. Suppes. Probabilistic inference and the concept of total evidence. March 23, 1966. (In J. Hintikka and P. Suppes (Eds.), Aspects of Inductive Logic. Amsterdam: North-Holland Publishing Co., 1966. Pp. 49-65.

95 P. Suppes. The axiomatic method in high-school mathematics. April 12, 1966. (The Role of Axiomatics and Problem Solving in Mathematics. The Conference Board of the Mathematical Sciences, Washington, D. C. Ginn and Co., 1966. Pp. 69-76.

(Continued on inside back cover)

ED 079980

# SELF-OPTIMIZING COMPUTER-ASSISTED TUTORING: THEORY AND PRACTICE

by

Ralph B. Kimball

TECHNICAL REPORT NO. 206

June 25, 1973

PSYCHOLOGY AND EDUCATION SERIES

"PERMISSION TO REPRODUCE THIS COPYRIGHTED MATERIAL HAS BEEN GRANTED BY

IM355

TO ERIC AND ORGANIZATIONS OPERATING UNDER AGREEMENTS WITH THE NATIONAL INSTITUTE OF EDUCATION. FURTHER REPRODUCTION OUTSIDE THE ERIC SYSTEM REQUIRES PERMISSION OF THE COPYRIGHT OWNER"

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

Reproduction in Whole or in Part Is Permitted for

Any Purpose of the United States Government

Copyright © 1973, by Ralph B. Kimball
All rights reserved

INSTITUTE FOR MATHEMATICAL STUDIES IN THE SOCIAL SCIENCES

STANFORD UNIVERSITY

STANFORD, CALIFORNIA

Table of Contents


| | |
|---|---|
| Acknowledgments | iv |
| Chapter 1. Introduction | 1 |
| Research Motivation | 1 |
| Descriptive Model of the Educational Experience | 3 |
| Desired Characteristics of the Tutor | 4 |
| Contributions | 5 |
| The Student-Subject Model | 6 |
| The Tutorial Strategy | 7 |
| Experimental Results | 9 |
| Chapter 2. Student Models | 12 |
| Problems and Solutions | 12 |
| Techniques and States | 14 |
| The Technique Choice and Technique Result Probabilities | 16 |
| The Markovian Assumption | 17 |
| The Solved and Give-up Trapping States | 18 |
| The Failure State | 20 |
| Identification of the Problem Solving States | 26 |
| State Selection Examples | 29 |
| Differential Equations Example | 32 |
| Physics Laboratory Example | 35 |
| The Transition Probability Matrix | 40 |
| The Information Updating Model | 42 |
| The Student Learning Model | 46 |
| Expected Number of Steps | 57 |

Chapter 3. The Tutorial Strategy 60

The Plan of the Tutor 62

Tutor Initialization: Dynamic Subject Scope 62

Selecting an Example: The Tutor-Student Distance 63

Choosing a Technique 67

Unusual Technique Threshold 67

Hint Generation 69

Applying the Technique 70

Problem Length Threshold 72

Finishing the Solution 73

Tutor Learning 74

Diagnosing Poor Problem Solving Practices 76

Summary 78

Chapter 4. The Methods of Integration Experiment 80

The Choice of Methods of Integration 80

Identification of the States and Techniques 81

The Tutor as Seen by the Student 84

Choosing the Problem 85

Choosing the Technique 86

Applying the Technique Chosen 86

A Sample Student-Tutor Dialogue 88

Implemetation of the Thresholds and Archives 93

The Information Updating and Student Learning Models 97

Description of the Computer Tutorial System 99

Chapter 5. Experimental Results 101

The First Experiment 101

The Second Experiment 102

Determination of the Failure Parameter 103

The Existence of Learning Discontinuities 104

The Student's Expected Number of Steps 111

The Probability of Entering a Failure State 114

Estimation of the Learning Parameters 117

Definition of a Learning Rate for Tutorial Systems 123

The Tutor Optimization Experiment 124

Conclusions 127

Recommendations for Future Research 128

Bibliography 130

Acknowledgments

This thesis would not have been possible without the inciteful prodding and friendly encouragement of Professor Richard D. Smallwood. His human approach to problem-solving combined with his standards of excellence have set an indelible example for his students.

I must also thank the Institute of Mathematical Studies in the Social Sciences, and particularly the Institute's director, Professor Patrick Suppes, for the generosity of providing the computational facilities for this three-year undertaking. Many people within the Institute provided valuable contributions to the effort, but particular thanks are due to Adele Goldberg, David Levine, and Robert Smith for discussions, advice, and hard work.

The National Science Foundation generously supported this research with a personal fellowship and with underwriting the basic computer costs under contract NSF-GJ443X.

Finally, the tutor experiment could not have succeeded without the eager and sometimes patient interest of my students, notably Ruth Cronkite and Nic Moll. The preparation of the final manuscript was made possible by the diligent typing of the author, Ralph Kimball, and the expert advice of computer-manuscript composer Robert Smith.

# Chapter 1

## Introduction and Statement of Contributions

### Research Motivation

The original goal of this research was to investigate ways to stimulate creativity using the computer as a medium. One of the criticisms of existing uses of computers in education has been that "programmed instruction" is an unimaginative application of conventional teaching practices to a computer learning environment. Thus my hope was to understand the basis of this criticism and to suggest a remedy.

I soon learned that there are actually three distinct areas of computer-education activity. One entire approach, which I call the "environmental" approach, is based on the assumption that the learner must discover nearly everything himself, without an over-riding structure to determine how new concepts are to be presented. In practice, this may mean that the student is introduced to a computational environment and told that he may explore in any direction he chooses. However, some measure of individual guidance is desirable, particularly if the student becomes confused or bored. In the environmental area this guidance has always come from a human teacher except in the trivial case of providing programming "diagnostics". Seymour Papert's work with computing environments for children is the most interesting example of the environmental approach.

The remaining two areas of activity embrace conventional computer assisted instruction (CAI). These areas have been called "frame oriented CAI" and "information structured CAI". Frame oriented CAI is based on prestored units (frames) of subject matter, ranging in size from individual sentences up to several paragraphs of text with associated drills. The interesting research problems have focused on finding strategies for presenting the frames to the student in some desirable way. These strategies usually involve a student learning model that predicts what will happen to the student upon exposure to the frame, and a dynamic programming scheme to find future sequences of frames to show the student. Although the student is restricted compared to the environmental approach, the predictability of both the frame content and the student responses allows the computer to deal with a larger domain of student aberrations. For instance, if the student fails to understand a frame, he can be shown an easier frame. Much of the ground work for frame oriented CAI was laid in a dissertation by Smallwood.

Information Structured CAI is the newest of the three areas, and as the name suggests, is a collection of techniques drawn mostly from artificial intelligence that exploit the structure of the subject being taught. These programs are often characterized by considerable student control of the dialogue as well as a major effort to give the learning episode a human, rather than computer, flavor. One well known example is Jaime

Carbonell's semantic information net that allows students to interview the computer about South American geography. His program has a large data base constructed in such a way that arbitrary questions can be answered and deductions can be drawn from disparate facts.

My original interest in stimulating creativity gradually evolved into a desire to isolate the moments of learning in the normal educational process, and if possible, to recreate these moments in a computer system. Since I had had experience teaching mathematics, it occurred to me that many students have valuable learning experiences in a tutorial situation, in which they return to the teacher for answers to questions arising from an initial attempt to understand the subject.

At this time I read two dissertations in artificial intelligence on the subject of methods of integration. The conclusion of the dissertations was that computers could solve integrals as well as an expert human integrator. I immediately wondered whether a tutor could be constructed for methods of integration that in the same sense was as good as an expert human tutor. This idea gradually developed into the subject of my thesis.

## Descriptive Model of the Educational Experience

An informal model of the educational experience shows the intended role of the tutor:

1. A Need to Know
2. The Teacher Explains (Lecture)
3. The Student Thinks (Problem)
4. The Dialogue (Tutorial)

In the fourth phase, the student has already had some exposure to the subject from the Lecture phase and has pondered some kind of synthesis process in the Problem phase. Thus the somewhat knowledgeable student returns to the more knowledgeable teacher for an interactive dialogue. The purpose of this description is to emphasize that the tutor is not used as a primary instructional medium, and that the general direction of the episode is up to the student.

## Desired Characteristics of the Tutor

A good human tutor

1. transmits problem solving heuristics
2. chooses appropriate examples
3. deals with arbitrary student examples
4. handles a wide range of student backgrounds

and 5. learns student heuristics if they are superior.

The goal of the research was to construct a tutor with these desired characteristics and in the process to establish a rationale for constructing tutors for subjects other than methods

of integration.

## Contributions

This research has produced six main contributions; it has

1. extended and clarified the definition of a tutor in computer-based education.

2. established a logical and quantitative methodology for transferring problem solving heuristics in a computer tutorial situation.

3. experimentally supported a model of how the student heuristics change when a failure is encountered.

4. defined a methodology for measuring learning in a tutorial situation and experimentally supported the model by showing a positive rate of learning on real students.

5. defined and implemented a scheme for tutor improvement.

6. combined the results of new research in symbolic integration and algebraic simplification for use in computer based education.

The research described in this thesis is of both a theoretical and experimental nature. The theoretical ideas can be summarized by the following:

1. For a structured subject, a tutorial system can be constructed that

   a. causes convergence of the student's heuristics to those of the tutor;

   b. chooses optimum examples, recommends solution scheme choices, and shows how to apply techniques;

c. accepts arbitrary examples;

d. allows measurements of the student's learning rate and expected number of steps to solution;

e. makes real time tutorial decisions on the basis of problem length, unusualness of approach, and overall problem solving trouble;

f. adjusts to different student backgrounds;

and g. learns student heuristics if they are superior.

The experimental section describes the following:

A computer based tutor for methods of integration was constructed to illustrate the theoretical claims. In addition,

a. algorithms were developed for substitution, integration by parts, partial fraction expansion, use of trigonometric identities, and completion of the square;

b. a preliminary experiment was run with four calculus students;

c. a main experiment was run with fifteen calculus students, the results of which are presented in Chapter 5.

## The Student-Subject Model

The ability of the tutor to understand the student's actions depends on a model of the student interacting with the subject. The principle components of this model are an exhaustive set of observable problem solving states and a set of

transformation techniques to transform from one problem to another. A sample student solution trajectory is shown in Figure 1.1 for the subject of methods of integration. The definitions of the states and techniques involved are discussed at length in Chapter 2.

Integral solving is modeled here as a Markov process. We also assume that when the student is confronted with a problem, his probability of responding with one of the techniques is drawn from a simple multinomial distribution. The parameters of this multinomial distribution are never known exactly by the tutor, and can only be inferred from the student's responses. The tutor applies a simple Bayesian inference process to its prior estimate of the student's response probabilities each time the student emits a response. This is called the information updating process.

If the tutor thinks the student has just learned a new approach, it will isolate the response probabilities it feels have changed the most and will apply what is called the student learning model to predict the effect of this change on future responses.

## The Tutorial Strategy

The tutor allows the student great latitude during the problem solving episode. The student is free to suggest his own example, pursue his own solution, and ask for help. The tutor

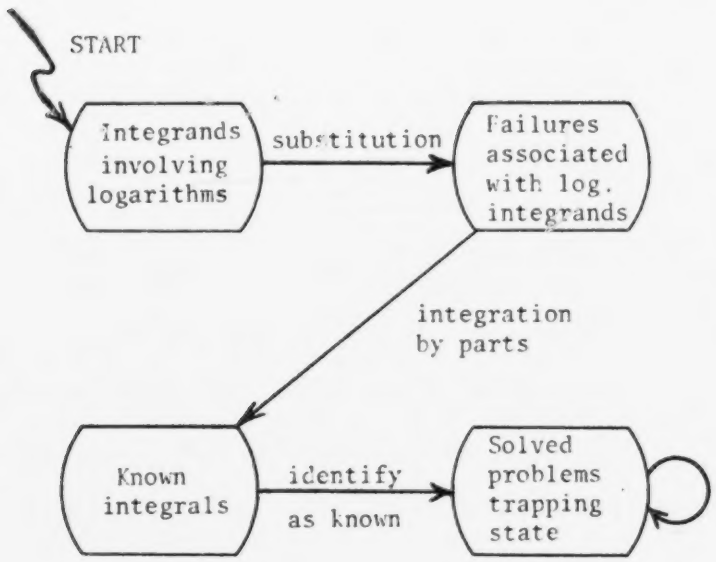


Figure 1.1 A sample student solution trajectory. The student began with the integral $\int X \log(X)\, dX$, tried the substitution $U = \log(X)$ which failed to yield a simpler integrand, returned to the original integral and successfully transformed it by integration by parts to $(1/2)\, X^2 \log(X) - \int X/2\, dX$. This "known" integral was then solved by inspection. A complete student protocol involving this problem is given in Chapter 4.

Figure 1.1 A sample student solution trajectory.

always understands what the the student is doing, however, and in unusual cases of trouble or misunderstanding will step in to ask the student if he needs assistance. The student can request three levels of help from the tutor: choosing a new problem; finding the right technique to apply; and applying a given technique the best way. In the first two cases the tutor relies completely on its archive of example problems. All recommendations are made by summarizing the actions taken in similar situations in the archive problem solutions. The third type of problem help depends on specific algorithms programmed into the tutor, and thus is known as "wired in heuristics".

After the student completes the problem, the tutor checks the student's overall problem solving patterns against its own. If unusual trouble areas are apparant, the tutor scans a special archive of problems for an example to show the student in a "forced response" mode. Finally, if the student has produced a superior solution for one of the tutor's archive problems, the tutor will incorporate his solution and will forget the old one. In this way, the tutor can significantly improve its own teaching performance.

## Experimental Results

A three week experiment with 15 students was conducted at Stanford University with students interested in sharpening their calculus skills. A total of 284 problems were worked, of which

258 were chosen from the problem archive by the tutor, and 282 successfully solved. On 90 of the problems tutorial help was requested.

Student response data indicated that the information updating and the student learning models could be considered separately since the observed response discontinuities seemed to be very closely related to the instances of entering failure states. 85% of all the response discontinuities occurred when the students entered failure states, although only 45% of the instances of failure states led to obvious response discontinuities. Furthermore, the observed frequency of the technique applied immediately before the failure occurred was reduced by a average factor of 0.17 over all the observed learning discontinuities. This was an important result for the student learning model since it singled out a particular technique as undergoing a dramatic change each time the discontinuity occurred.

On the average, each student made 0.101 fewer additional steps per problem worked than the tutor. In other words, if a student tended to work every problem using 3.0 more steps than the tutor initially (a typical rate), then after 15 problems he took on the average only 3.0 - (15 * 0.101) = 1.5 more steps than the tutor to work each problem. We call this the student's convergence rate.

On the average, each student made 0.011 fewer failures per problem step. Thus if the student averaged 0.5 failures per step

initially (a typical rate), then after 15 problems (a total of perhaps 40 steps) he made 0.5 - (40 * 0.011) = 0.1 failures per step.

Of the 73 original archive solutions, no less than 18 were improved by the students. This was a major surprise to the author, since he considered himself an expert integral solver. The tutor's expected number of steps to solution decreased 1.38 steps (20.3%) for quotients of polynomials, 1.150 steps (25.7%) for fractional powers of polynomials, and 1.00 steps (33.3%) for fractional powers of trigonometric functions. In the quotients of polynomials case, the tutor acquired solution schemes that had never occurred to the author.

# Chapter 2

## Student Models

The student-subject model forms the foundation of the tutorial system. Once we have established a methodology for treating the student, the development of the tutorial strategy in the next chapter follows naturally. The basis of the student-subject model is a scheme for classifying the subject matter. This chapter will introduce a Markovian formalism to describe the student's dynamics. We shall introduce the notions of "problem" and "solution" and shall show that the student belongs to one of an exhaustive set of problem solving states at all times. The important tutorial concept of the student's failure state will be discussed and examples of state definitions for several subjects will be given. We shall introduce a simple Bayesian scheme to update our knowledge of the student's problem solving patterns from his responses. From this we can calculate interesting quantities relating to the student's performance such as the mean and variance of the student's expected number of steps to solution.

### Problems and Solutions

We begin with the notions of "problem" and "solution". Since we have dealt practically with well defined subjects where there are established limits, we allow the argument to proceed

informally at this stage to avoid introducing abstract questions about the nature of subject matter and in what way statements are consistent with subject matter.

By "problem" we refer to some statement that we wish to reduce to a form that warrants no further reduction. A tutorable "subject" consists of a space of possible problems defined by a set of problem construction rules, plus a set of transformation techniques used for reducing the subject's problems. Each time a transformation technique is applied to a problem, a new problem is generated. As we shall see, this process can go on indefinitely until the student either gives up or the problem requires no further reduction. In the latter case, we call the record of successive problems and applied techniques a "solution" of the original problem. For example, we shall discuss in Chapter 4 the subject of methods of integration in which the problems are indefinite integrals and a solution is a sequence of ordered pairs of the form

(problem , technique).

Later in this chapter these ideas are applied to a number of sample subjects: elementary arithmetic; the solution of differential equations; and a simulated physics laboratory. In addition, we discuss briefly the subjects of medical diagnosis and electronic trouble-shooting.

Techniques and States

An important property of a solution is that each successive problem is generated from the previous problem by the application of one of a set of transformation rules called "techniques". We designate the set of problem transforming techniques by $\{T_j \mid 1 \leq j \leq N_T\}$ where $N_T$ is the allowed number of different techniques for this subject.

Of prime interest to a tutor is how the student decides on a given technique when presented with a problem. It is natural to suppose that different problems will elicit different choices of techniques. An important parameter of the student is his probability of choosing a given technique in different problem solving situations. This section introduces the definition of the student's technique choice probability. Looking at this choice probabilistically, we say that the probability $t_j$ that a student will choose technique $T_j$ is dependent upon the particular problem. Because all of the subjects investigated have an infinite number of possible problems, it is inconvenient to index the technique choice probability $t_j$ by the specific problem. Furthermore since we shall usually be interested in estimating what the student's technique choice probability will be for some new problem, we shall find it convenient to assume that at all times the student occupies one of a set of mutually exclusive and exhaustive states of the problem solving process. We now speak of the probability $t_{ij}$ that the student will choose technique $T_j$ given that he

occupies the $i^{th}$ problem solving state.

It is natural to define the states of the problem solving process in terms of a description of the problem the student is working on. We insist that the parameters of a given state characterize the student in all problem solving situations involving problems fitting a certain description. Thus the state parameters amount to an encoding of our observations of the student's past history and our prior estimates of his problem solving patterns.

For tutorial purposes we often are interested in separately encoding our information about the student after he has applied a technique unsuccessfully to a problem. In particular it would be embarrassing to claim that the probability of the student choosing technique $T_j$ was some fixed quantity regardless of whether the student had tried the technique unsuccessfully on the previous step. Thus we introduce the concept of a "failure state" for each class of problem description. The properties of these states are discussed later in this chapter. Thus the set of problem solving states will be defined by

$$\{s_i \mid 1 \leq i \leq N_S\} = \{\text{problem description states}\} \cup \{\text{student failure states}\}$$

where $N_S$ is the total number of states. We intend that the states $\{s_i\}$ are disjoint, and that they will span the set of possible circumstances the student can be in.

For practical reasons the states of the process must be both observable and finite in number. Observability in this context means only that we shall choose state definitions beforehand that will guarantee an unambiguous identification of the student's state at all times. The motivation for complete observability arises from the need to compare the state of the student with a similar state established by the tutor. In general, the parameters of a given student state (such as $t_{ij}$) may be imperfectly known. The restriction to a finite and reasonably small number of states is not a major assumption since under certain conditions we can agglomerate a possibly infinite number of seldom encountered states into a single "other" state. Typically a subject will have at least one problem-description "other" state and one failure-defined "other" failure state. In practice, the requirement that the number of states $N_S$ be reasonably small means only that an $N_S$ by $N_S$ matrix can be inverted off-line from the tutorial episode without unreasonable cost.

## The Technique Choice and Technique Result Probabilities

The most important parameter of the student's state $s_i$ is the technique choice probability $t_{ij}$ that he will choose technique $T_j$ given that he is in state $s_i$. The technique choice probabilities, if perfectly known, would be a complete map of the student's problem solving heuristics. Even this map, however,

would not suffice to determine the student's expected solution of a given problem. We must assume that some students are more adept than others at applying a given technique to a problem, thus giving rise to an uncertainty, as to what new state will result from the transformation. We will define $q_{ijk}$ to be the probability that given a problem in state $s_i$ and choice of technique $T_j$ the resulting state will be state $s_k$. We can define a state transition probability $p_{ik}$ from state $s_i$ to state $s_k$ by

$$p_{ik} = P\{s_k \mid s_i\} = \sum_j t_{ij} q_{ijk}$$

The Markovian Assumption

Up to this point we have avoided the question of whether the probabilities $t_{ij}$ and $q_{ijk}$ are dependent upon the past history of the solution being attempted by the student. True changes in these quantities do take place when the student gains a new insight into the problem solving process. However, useful general analysis of a history dependent process is immeasurably more difficult than if the transition probabilities are assumed to be dependent only upon the present state of the system. Bearing in mind that we are making an assumption, we shall proceed as if the student's technique choice probabilities do not alter until he verifies his new methods by completing the problem successfully. In other words, we make the Markovian assumption that the state

transition probabilities depend only on the present state of the system. It should now be clear why the failure states were introduced, since without them, the Markovian assumption is untenable. This allows us to remove the implicit dependence of $t_{ij}$ and $q_{ijk}$ on the previous trajectory of the process and to use the extensive theory of Markov processes in the analysis of the student's problem solutions.

## The Solved and Give-up Trapping States

We have made few general statements about how a student's problem solving state is defined, prefering to deal with the question for each of the examples to be shown. However, it is useful to define two states that occur in all problem solving domains that can be described by our Markov model. The first is the "solved" state. A student reaches the solved state when the problem has been transformed into one of a class of configurations that the tutor and the student have previously agreed to call "solved". In numerical problems the solved state corresponds to situations in which the equations have been reduced to one or more numbers. In formal proofs the solved state is reached when the desired result follows directly from a previously established theorem or axiom. In methods of integration the solved state is a class of "known" integrals, such as $\int X\,dX$ or $\int \sin(X)\,dX$.

The other canonical problem solving state is the state of "giving up" on a problem. At this point the student has reached

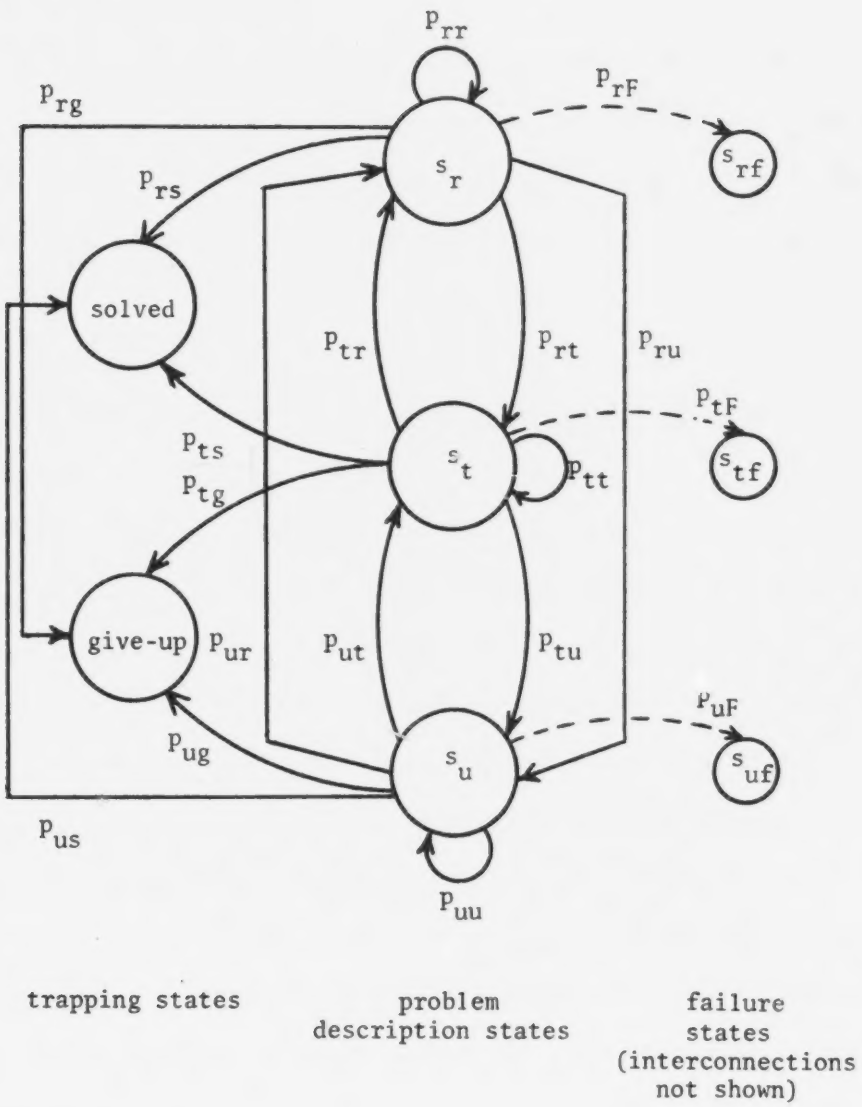


Figure 2.1. A three state problem solving model

a terminal situation and his transition probabilities to other states are all zero. In fact, we recognize both of these states as trapping states of the problem solving process since once the student enters one of them he makes no more transitions to other states. Figure 2.1 illustrates the problem solving model for an arbitrary three state example.

## The Failure State

A basic assumption we make about problem solving behavior is that a student will try in succession all of the applicable techniques he knows until the problem is completed. (We allow "giving up" to be a legal technique to complete a problem). It is true, however, that once a student has attempted a technique on a problem unsuccessfully he does not return to the state where application of the technique is still as likely. But it is also true that the student's new state is closely related to the old state. We expect that the student's response probabilities are substantially the same after a failure except for the technique that failed. We now pose a simple mathematical model to represent this effect. If we choose a "failure parameter" $\theta$ where $0 \leq \theta \leq 1$ and $\theta$ is expected to be small, then we assume that the new technique choice probabilities, subject to a failure of technique $T_j$, are

$$t_{ij,posterior} = \theta t_{ij,prior}$$

$$t_{im,posterior} = \omega t_{im,prior} \quad , m \neq j$$

where

$$\omega = \frac{1 - \theta t_{ij}}{1 - t_{ij}} \quad \text{(in terms of prior probabilities)}$$

renormalizes the posterior probabilities to 1. It is possible that even the second technique chosen may be unsuccessful, thus generating another set of technique choice probabilities modified by the failure parameter $\theta$.

In the tutor-student model, the failure state assumes an important role because we assume that the student is reassessing his own technique choice probabilities in order to transform the problem. Experimentally, we observed that in approximately 50% of the cases where the student entered the generalized failure state (to be discussed), a significant change occurred in his technique response probabilities.

A complication arises from our lack of a priori knowledge of which failing technique the student may choose. To justify the Markovian assumption (that the properties of the system do not depend on the past history of the process), we would have to specify in advance all the possible failure states that could arise.

In Figure 2.2 we show a system with two problem classification states, $s_a$ and $s_b$. A small subset of failure

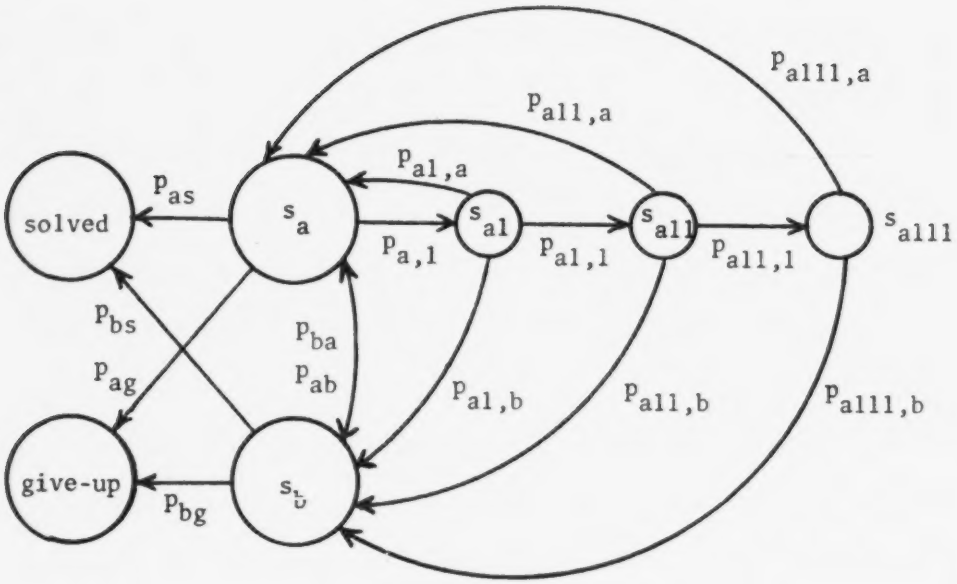


Figure 2.2. A failure network for a two state system.

states for state $s_a$ are shown. State $s_{a1}$ corresponds to a failure of the #1 ranked technique by a student occupying state $s_a$. State $s_{a11}$ corresponds to a failure of the #1 ranked technique for state $s_{a1}$, and so on. States such as $s_{a1111}$, $s_{a2}$, $s_{a3}$, $s_{a12}$, and $s_{a234}$ are not shown.

If we insist on the exact Markov process formalism, we must include an infinite number of failure states with each problem description state. To avoid this, we suggest two approaches to collapse this multitude of failure states into a single state.

In the first approach we generalize the problem solving process from a discrete-time Markov process to a discrete-time semi Markov process. Now not only do we associate a transition probability $p_{ij}$ from state i to state j, but we also specify a holding time $\tau_{ij}$ that is a delay the process experiences in state i before making a transition to state j. The holding time $\tau_{ij}$ is described by a holding time mass function $h_{ij}(m)$ defined over all integral values of time from m=0 to m=infinity. Referring to Figure 2.2, we see that the probability of entering the "failure network" from state $s_a$ is $p_{a,1}$. Immediately after entering the network, the probability of eventually returning to state $s_a$ is

$$p(s_a|\text{failure}) = p_{a1,a} + p_{a1,1}\,(p_{a11,a} + p_{a11,1}\,p_{a111,a})$$

The holding time mass function is then

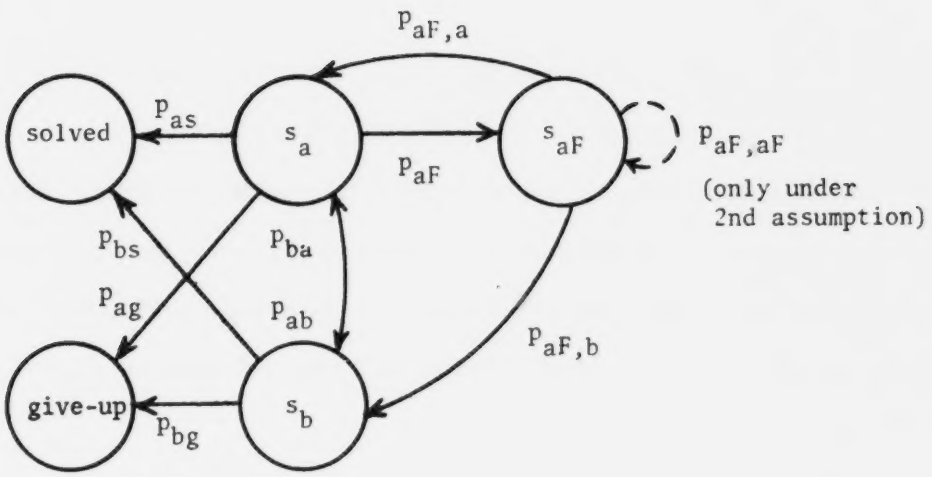


Figure 2.3. The collapsed failure network.

$$h_{fail,a(m)} = \frac{p_{al,a}}{1 - p_{al,b}} \delta(m-1)$$

$$+ \frac{p_{al,l}}{1 - p_{al,b}} \left\{ \frac{p_{all,a}}{1 - p_{all,b}} \delta(m-2) + \frac{p_{all,l}}{1 - p_{all,b}} \delta(m-3) \right\}$$

Similar relationships hold for for $p(s_b \mid \text{failure})$ and $h_{fail,b}$. Thus if we are willing to cope with the additional generality of the semi Markov process we can collapse the entire structure down to that shown in Figure 2.3. Fortunately, we do not pay substantial additional penalties when calculating quantities of interest such as the mean number of steps to trapping, or the time-interval transition probabilities.

The second approach to dealing with failure states is an approximation, the exactness of which will not be discussed here. In this case, we modify our definition of "transition" in the problem solving process to allow only one step in the agglomerated failure state of Figure 2.3. Formerly we identified a transition in the process with the application of a technique. This remains unchanged in the new approximation except that in the failure state, only a successful application of a technique gives rise to a transition. The problem with this approach is that the Markovian formalism is in serious jeopardy if we believe the discussion of the previous section that the student's response probabilities depend on what kind of failure he made. Fortunately, this discussion is not central to the issue of

developing a tutorial strategy.

Summing up, the failure states can be modeled by either 1) a semi Markov formalism with attendant additional complexities; or 2) an approximation which changes the definition of transition from the failure state and either a) raises doubt about the validity of the Markovian assumption, or b) tempts us to ignore the response probability model of the previous section. In Chapter 5 we shall have more to say about whether the reponse probability model is supported by experimental evidence.

## Identification of the Problem Solving States

The most important step in developing a tutor for a subject is identification of the problem solving states. Although there is no known algorithm for selecting an optimal set of states, we shall outline a two step procedure that has been successful in practice. The first step is to isolate the main stages of problem solving in the new subject. We call these stages decision points. The intention is that separate decision points must involve a separate set of alternatives for the student for which he must employ his problem solving judgement to proceed. Some subjects may only involve one decision point. The second step of state selection involves selecting a set of problem description classes for each decision point. These classes define all the possible problem descriptions for the student when he reaches that decision point. The remaining states, the failure

states, must also be defined, depending upon how the teacher wishes errors to be handled by the tutor.

The state structure is highly variable. For a subject like methods of integration the problem solutions can be thought of as a single decision point with perhaps thirty distinguishable states (representing fifteen problem description states and fifteen failure states). After an integration problem is transformed, either the problem is solved or the student returns to the same decision point, characterized by the question: how can the integral be transformed to yield a simpler integral? The state structure for integration contains only one decision point since interconnections among the various problem types are quite general. The student has substantially the same set of subalternatives for each problem. In other words, an "exponential integrand" may be transformed into a "quotient of polynomials" or into a "trigonometric integrand" or into an "exponential integrand" again.

The second step of our procedure, that of choosing problem description states at each decision point, is more difficult. The difficulty stems from the fact that in many cases the most natural way to select problem descriptions is according to the method of solution. We soon discover that 1) different people solve the same problem in different ways and 2) students new to the subject have no way to tell what state they are in. It is no help to be told that "this kind of problem is solved using technique T" when

"this kind of problem" is defined as the set best solved by technique T. As a guiding principle we state the following:

> A problem description state must not be defined solely by means of the technique used to solve its member problems unless those problems are easily recognized and the use of the solving technique is the sole recommended practice of experts in the subject.

It could be argued that if a state is defined that meets the exception conditions of the above principle then since no judgement is required, it is really unnecessary to have a decision point at that stage of the problem solving process. However we shall see in the example of methods of integration that often there exists a mixture of technique dependent states and states defined from other considerations.

Our main interest in defining the problem description states is that these state definitions must not depend upon the observer. We shall often wish to compare a student's state with a similar one established by the tutor because the tutor at all times knows the techniques it would use to try solving a problem of the given type. It is essential that in such a case we are comparing the same subset of the total problem domain. This is why we must take pains to allow technique dependent states only when an impartial observer would agree that the definition was natural.

The definitions of the other problem description states depend on how problems are described. Roughly speaking it is desirable to group together problems with similar characteristics. Surprisingly, such groupings may yield a rich variety of different problem approaches. For instance, the integrals

$$\int \frac{1}{1+x^2}\,dx \quad , \quad \int \frac{1}{1+x^3}\,dx \quad , \text{ and } \quad \int \frac{x}{1+x^2}\,dx$$

might all be relegated to the same problem solving state since they have similar characteristics, but they are usually solved by quite different approaches.

State Selection Examples

Because of the importance of the selection of states to the tutorial strategy, the following examples are presented of the state selection procedure as it could be applied to familiar school subjects.

To emphasize a point we shall choose first the simple subject of multiplication of integers, as might be learned by elementary school children. In this subject we will identify only one decision point, which can be described by the question: what is the product of the integers m and n? This subject can be represented by a simple problem structure consisting of one problem description state and one trapping state corresponding to a

correct solution. Although the subject of integer multiplication presents us with no automatic state divisions for its single decision point, the students provide considerable additional state structure through various failure mechanisms. Specific remedial action can then be taken to deal with the problems of each failure type. For example, let us divide the students' possible erroneous responses into the following four states:

1. Differs from the correct answer by either 1 or 2
2. Differs from the correct answer by
   a multiple of m or n
3. Null response
4. Other incorrect number

We have tried to construct states as dependent as possible on specific "failure techniques" that might arise. For state #1 above we have assumed that the student has memorized the multiplication tables imperfectly. For state #2 we assume that the student confused one entry in the table with another. State #3 represents a lack of understanding of the multiplication process by the student. Figure 2.4 shows the state transition diagram for this situation.

Many of the ideas developed in this chapter are of interest even in such a simple case as our multiplication example. The expected number of steps $\nu$ for each student is a measure of the expected number of errors he will generate on a typical

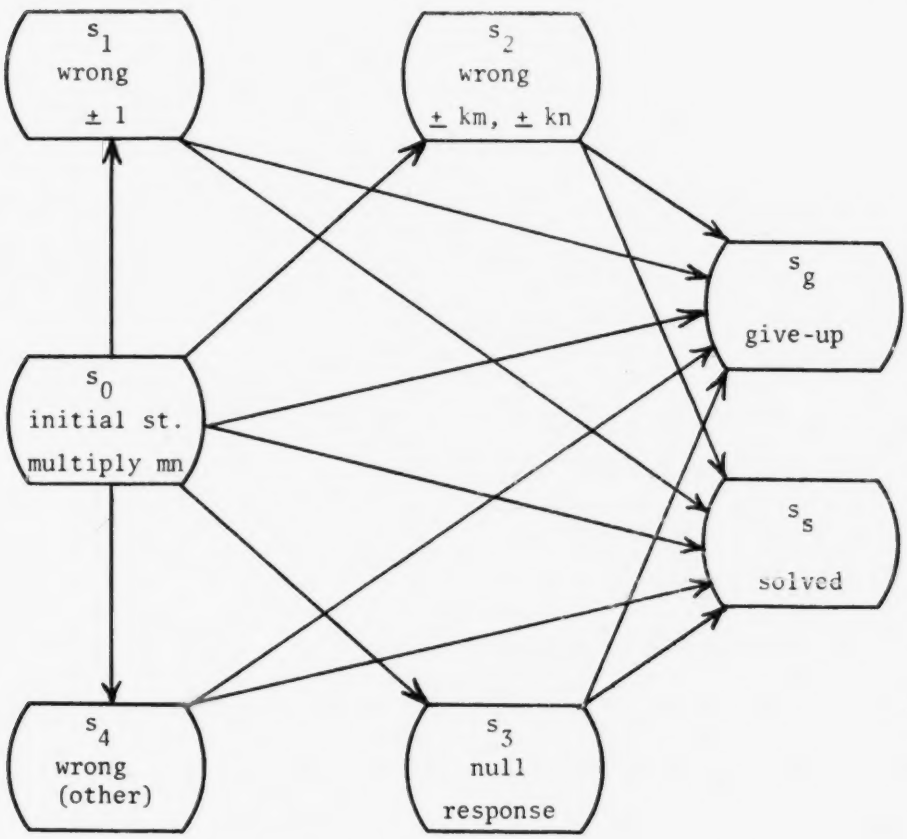


Figure 2.4. State structure for elementary multiplication.

multiplication problem. Probabilities like $p_{1s}$ (the probability of a transition from state 1 to state s) are measures of how probable a correct response is after a student makes an error of a certain type. The change of this probability over time would be a measure of the tutor's effectiveness. Of course we must remember that the state transition probabilities depend upon both the technique choice probabilities and the technique result probabilities; that is,

$$p_{ik} = \sum_j t_{ij} q_{ijk} .$$

Several of the $t_{ij}$'s may be non-zero for a given state $s_i$. For example, even though we find the student in state $s_2$ of our multiplication process, we can only surmise that the particular proposed failure is likely. It could happen that the student made a wild guess that fit the requirements of state $s_2$!

Differential Equations Example

A more complicated example can be constructed from the subject of ordinary linear differential equations. Apart from the increased complexity of the problems compared to integer multiplication, we must introduce three new generalizations in the problem solving process. The first is the added multiplicity of problem description states. These include first order equations, second order equations with constant coefficients, and second order equations with variable coefficients. The second is the

possibility of two decision points in the process rather than one as in the arithmetic example. These decision points correspond to finding the particular solution and finding the homogeneous solution. The third new complexity is that the process of solving differential equations itself often involves a fundamental uncertainty to the approach. Once a person learns how to multiply integers, he usually approaches multiplication problems algorithmically. However, differential equations cannot be solved by any comparatively simple algorithm. The student must exercise his judgement at each decision point to decide which type of solution scheme to pursue. The power of the tutorial methods we are developing allows us to consider not only algorithmic subjects like integer multiplication but especially subjects where problem solution is something of an "art". A preliminary state transition diagram for ordinary linear differential equations of the first and second orders is shown in Figure 2.5 along with some interesting failure states and some indicated technique possibilities.

As in the multiplication example, the expected number of steps allows us to calculate the expected number of errors made by the student. If $\nu_1$ is the expected number of steps to solution for first order equations, then the number of errors for each starting state is

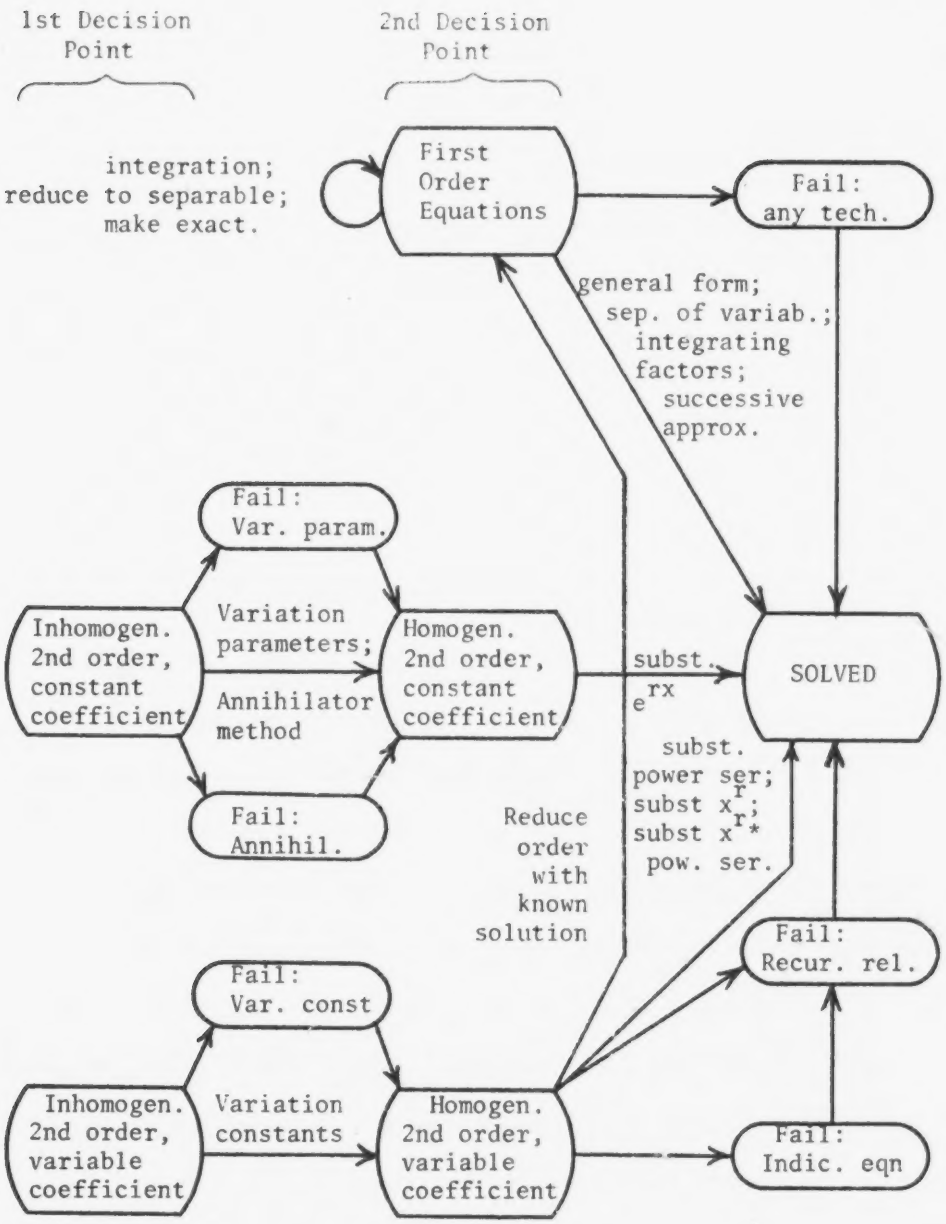


Figure 2.5. State structure for differential equation solving.

$$E_1 = \nu_1 - 1$$

$$E_{h2cc} = \nu_{h2cc} - 1$$

$$E_{i2cc} = \nu_{i2cc} - 2$$

$$E_{h2vc} = \nu_{h2vc} - 2*p_{h2vc,1} - (1 - p_{h2vc,1})$$

$$E_{i2vc} = \nu_{i2vc} - 3*p_{h2vc,1} - 2*(1 - p_{h2vc,1})$$

where "i2vc" refers to the state labeled inhomogeneous $2^{nd}$ order variable coefficients. In the first three equations we subtract either 1 or 2 steps corresponding to error-free solutions. In the last two equations we subtract off the number of error-free steps weighted by the probability that the student will reduce the equation to first order. This model has employed three different types of failure states. For inhomogeneous second order equations with constant coefficients there are two failure states depending upon the method of solution attempted by the student. For homogeneous second order equations with variable coefficients there are two failure states corresponding to steps common to more than one of the techniques. Finally, for first order equations we have condensed all the failure states into a single state because of the larger number of possible solution techniques. Each of these failure state examples may be handled by the failure state coalescence techniques discussed earlier in the chapter.

## Physics Laboratory Example[1]

The third example of choosing states is a simulated

physics laboratory. Imagine that the assignment is to measure the acceleration of gravity. The students are performing this experiment as part of a series of simulated mechanics experiments. In fact, if desired, the students, rather than simulating the experiment, could actually do the experiments and allow the tutor to directly observe the measurements. The laboratory has at its disposal a variety of mechanical devices: spring mass systems, adjustable dashpots, falling bobs, pendulums, inclined planes, cubes and spheres of specified coefficients of friction, and assorted pulleys, levers and motors to suit the student's needs. The student also has a number of measuring devices available including a stop watch to measure time, a variety of rulers and micrometers to measure distance, and a recording device that measures position at frequent fixed intervals of time from which instantaneous velocity can be estimated. The students are then allowed to experiment any way they desire, as long as they make a careful estimate of the measurement errors incurred by their particular setup. If their error on any particular measurement is too large, or if the tutor thinks that the student's measurement technique is inferior, the student must try a new approach.

The constraints we have just outlined are fairly typical of college physics laboratories, yet this situation lends itself

---

1. Compare for example the EXPERIMENT program for the PLATO CAI system by Bitzer, Probst and Walker.

well to the problem solving model. We consider the student to be in one of three problem description states at all times, unless he is one of the failure states or the giveup or solved states. These three states correspond to an attempted measuring of time, distance, and instantaneous velocity. Briefly, we can imagine several approaches to this problem: 1) measure the instantaneous velocity of the falling bob, solving $g = v/t$ ; 2) Measure the period T and length L of the pendulum, solving $g = 4\pi^2T^2/L$ ; 3) Measure the instantaneous velocity and vertical distance traveled of the pendulum or the falling bob, solving $g = v^2/(2d)$ ; 4) Measure the time and distance traveled of the falling bob, $g = (2d)/T^2$ ; 5) measure the instantaneous velocity of a solid sphere rolling down an inclined plane of length L and inclination angle $\psi$ , measuring v and L, solving $g = v^2/(5*L*\sin(\psi))$. Each of these approaches involves a succession of measurement techniques. When the correct measurements are all made, the problem is solved. Figure 2.6 shows the state model of our physics experiment.

This example has a particularly rich problem solving structure. Transitions from each measurement state to any of seven other states are possible. The successful transitions (to non-failure states) may be due to any of several measurement techniques applied by the student. In this example the technique result probabilities (the $q_{ijk}$'s) play a major role since a measurement might not be made with the accuracy necessary. Notice that the failure states may be entered from any of several states.

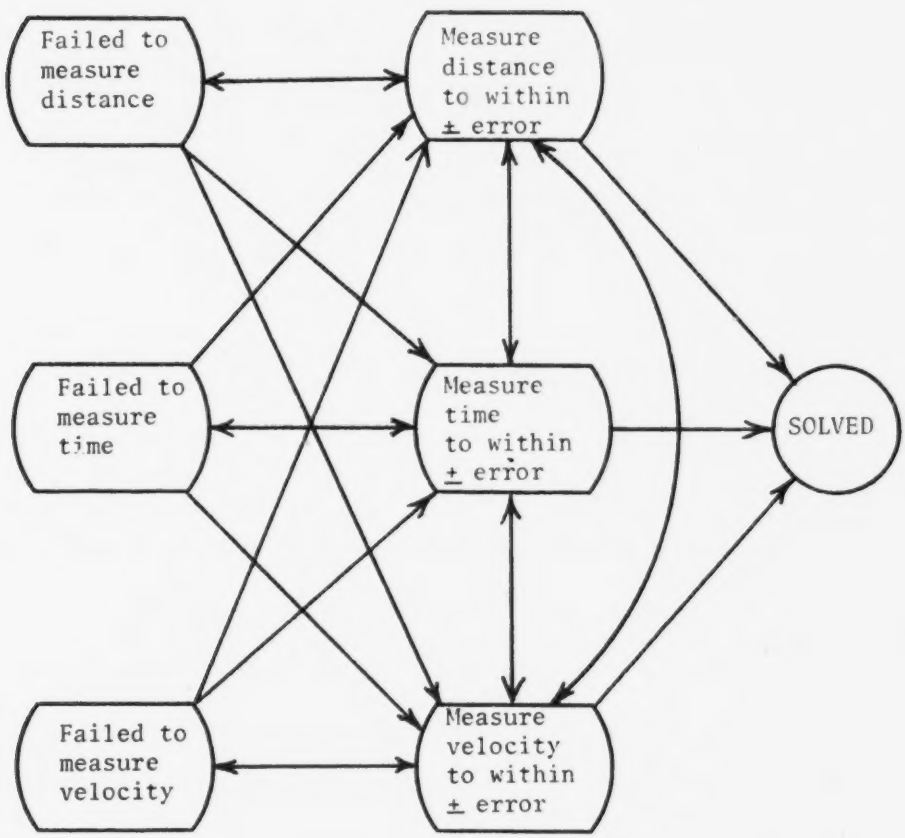
Failed to measure distance
Measure distance to within ± error
Failed to measure time
Measure time to within ± error
SOLVED
Failed to measure velocity
Measure velocity to within ± error

Figure 2.6. State structure for a simulated physics laboratory.

In this case they each represent the failure to apply one of a class of measurement techniques. As in the other examples, the expected number of steps is a good measure of a student's overall expertise. The minimum number of steps of solution is two; thus extra steps represent failures to measure the quantities accurately enough or failures to set up the measurements correctly.

The state definitions in this example were chosen so that all of the proposed mechanics experiments could fit into one scheme. We can thus interpret the transition probabilities to the failure states as general weaknesses in the student's laboratory technique. Of course, the example is very simple, but the underlying ideas could be applied to a variety of other fields.

Finally, the subjects of medical diagnosis and electronic troubleshooting should be mentioned. In both of these subjects human judgement is required to solve problems. The doctor can be imagined to progress through various states of information when making a diagnosis (Wortman, 1972); at each state evaluating the available options and proceeding on the basis of his best judgement. The electronics troubleshooter also proceeds through states of information and must use his judgement to progress through the tree of all possible actions. These are examples of subjects well suited to the tutorial approach and to the student being in control of the dialogue as he moves from one decision point to the next.

## The Transition Probability Matrix

The transition probability matrix for the system shown in Figure 2.7 is

$$
\begin{array}{c|cccccccc}
s & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
g & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
r & p_{rs} & p_{rg} & p_{rr} & p_{rF} & p_{rt} & 0 & p_{ru} & 0 \\
r_F & p_{r_Fs} & p_{r_Fg} & p_{r_Fr} & p_{r_FF} & p_{r_Ft} & 0 & p_{r_Fu} & 0 \\
t & p_{ts} & p_{tg} & p_{tr} & 0 & p_{tt} & p_{tF} & p_{tu} & 0 \\
t_F & p_{t_Fs} & p_{t_Fg} & p_{t_Fr} & 0 & p_{t_Ft} & p_{t_FF} & p_{t_Fu} & 0 \\
u & p_{us} & p_{ug} & p_{ur} & 0 & p_{ut} & 0 & p_{uu} & p_{uF} \\
u_F & p_{u_Fs} & p_{u_Fg} & p_{u_Fr} & 0 & p_{u_Ft} & 0 & p_{u_Fu} & p_{u_FF}
\end{array}
$$

This is a stochastic matrix since all elements are greater than 0 and each row sums to 1. This corresponds to the fact that the states completely describe all possible student situations.

We could use the transition probability matrix directly to assign probabilities to the student being in given states n steps after starting in a particular state. However, this particular use of the transition probability matrix is of little interest in a tutorial system. We usually know which state the student is in and have a relatively vague idea of the trajectory by which he arrived there! In other words, our interest is in the student's parameters as ends in themselves. Not only do we want to determine the student's $p_{ij}$'s, but we are especially interested in the technique choice probabilities (the $t_{ij}$'s) that we consider

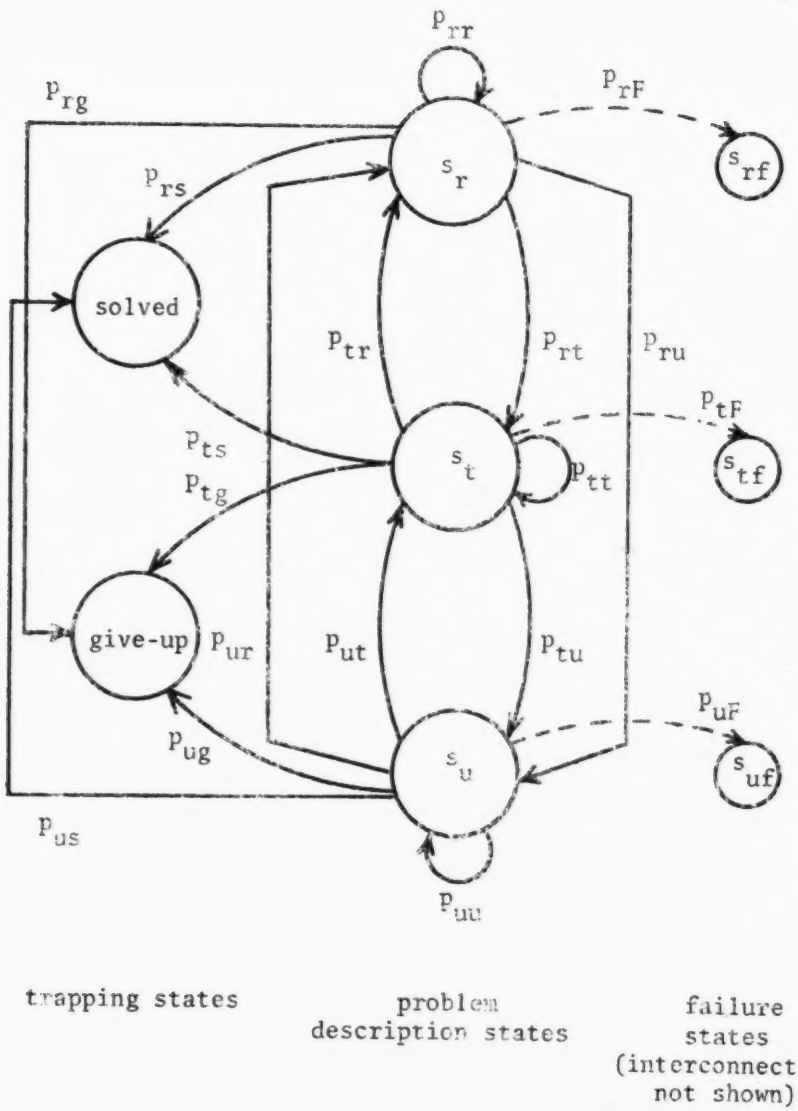


Figure 2.7. A general three state problem solving process.

to be the primary elements of the student's problem solving heuristics.

## The Information Updating Model

It is obvious that as observers we can never know a student's transition probabilities or his technique choice probabilities exactly. Two processes are usually working simultaneously that affect our observations of the student's parameters. The first is the number of responses measured at any given time. As we measure successive student responses, we must steadily update our model of the student. The other process, which we hope is working when the student is interacting with the tutor, is the learning process. In this case the student's real parameters are changing, not just our knowledge of them. Models will be developed to cover both situations as part of our effort to decide among alternative tutoring strategies.

We shall assume that the tutor's updating and the student's learning processes can be separated so that they operate independently. We shall now discuss a Bayesian approach to the updating process, leaving the learning process until the next section.

Consider a random variable $x_i$ defined on the possible outcomes of a student's selection of a technique given that he is in state $s_i$. We let $x_i = j$ if $T_j$ was the technique chosen, $1 \leq j \leq N_T$. We now let $t_{ij}$ equal the probability that $x_i = j$.

This is the formal definition of the technique choice probability $t_{ij}$.

If E is the event that in n independent selections from state $s_i$ the student chose technique $T_j$ a total of $n_j$ times, the probability of this event is given by the multinomial distribution:

$$\{E \mid t\} = \frac{n!}{n_1!\, n_2! \ldots n_N!}\, t_{i1}^{n_1}\, t_{i2}^{n_2} \ldots t_{iN}^{n_N}$$

where the notation $\{E \mid t\}$ means the conditional probability of the event E given the set of technique choice probabilities $t_{i1},\ldots,t_{iN}$.

The above assumes that the probability $t_{ij}$ is known. Since we are uncertain about its value, it is convenient to consider $t_{ij}$ itself as a random variable. To encode our uncertainty about $t_{ij}$, we place a prior distribution over the domain of possible probabilities.

For this purpose, and because of its convenient mathematical properties, we shall choose the Dirichlet distribution (also called the multidimensional beta distribution). In particular, the kernel of this distribution has the same form as the multinomial distribution (the conjugacy property). This allows the Bayesian modification of this distribution to be carried out in a very simple way. Formally, suppose the $t_{ij}$'s are

a priori jointly distributed according to

$$\{t|E\} = f_{\beta i}(y_1, y_2, \ldots, y_N | m_1, m_2, \ldots, m_N)$$

$$= \frac{(m_1 + m_2 + \ldots + m_N + 1)!}{m_1!\, m_2! \ldots m_N!} y_1^{m_1} y_2^{m_2} \ldots y_N^{m_N}$$

where $0 \leq y_k \leq 1$ and $\sum_k y_k = 1$.

The above distribution represents the technique choice probabilities in a particular state $s_i$; the subscript i being supressed. The N constants $m_1, m_2, \ldots, m_N$ are the parameters of this distribution and provide the encoding mechanism for all of our knowledge about the $t_{ij}$'s. An important property of this distribution is that the expected value of $t_{ij}$, is

$$E(t_{ij}) = \frac{m_j}{\sum_k m_k}$$

where again we have surpressed the subscript i. Other quantities such as the marginal distribution of a specific $t_{ik}$, the variance of $t_{ik}$, and the covariance of $t_{ir}$ and $t_{is}$ are easily derived. The key feature of the inference process is that the m-parameters will change as we obtain new data about the student. What parameters do we start with in the absence of knowledge about a particular

student? The choice of reponse probabilities characterizing the student before he makes a response is called the set of prior probabilities. Although in many processes the original, or zeroth set of prior probabilities will have little effect on the eventual characterization of the student, caution must be exercised in choosing this set since it will have a large effect on the initial tutoring strategy. Chapter 4 will discuss a specific choice of a zeroth set of prior probabilities for the subject of methods of integration.

We will now use Bayes' theorem to calculate the new distribution over the $t_{ij}$'s after event E has changed our knowledge of the student. Event E corresponds to the student emitting $m_i$ responses of type i, where i ranges from 1 to N = N. We have

$$\{T \mid E\} = \frac{\{T\}\{E|T\}}{\{E\}}$$

$$= \frac{f_{\beta i}(y_1,\ldots,y_N|m_1,\ldots m_N)\ \dfrac{n!}{\prod_j n_j!}\ y_1^{n1}\ldots y_N^{nN}}{\int_0^1\int_0^1 f_{\beta i}(y_1,\ldots,y_N|m_1,\ldots,m_N)\ \dfrac{n!}{\prod_j n_j!}\ y_1^{n1}\ldots y_N^{nN}\ dy_1\ \ldots\ dy_N}$$

Note that the denominator integrates to a constant.

$$= \frac{y_1^{m_1+n_1} \; y_2^{m_2+n_2} \cdots y_N^{m_N+n_N}}{\dfrac{(m_1+n_1)!\;(m_2+n_2)!\;\ldots\;(m_N+n_N)!}{(m_1+n_1+m_2+n_2+\ldots+m_N+n_N+1)!}}$$

$$= f_{\beta i}(y_1, y_2, \ldots y_N \mid m_1+n_1, m_2+n_2, \ldots, m_N+n_N)$$

The above steps illustrate the value of the Dirichlet form for the prior distribution of the $t_{ij}$'s. Bayes modification of the distribution requires merely adding to the exponents of the respective $y_j$'s the number of selections of technique $T_j$ during event E. The posterior expected value of $t_{ik}$ becomes

$$E(t_{ik}) = \frac{m_k + n_k}{\sum_j m_j + n_j} .$$

As we accumulate more responses from the student, we reduce the dependence of $E(t_{ik})$ on the initial choice of the m-parameters, until in the limit of an infinite number of responses they have no effect.

## The Student Learning Model

Earlier we remarked that two processes caused our state of information about the student to change. First we developed a Bayesian scheme for calculating the reduction of our uncertainty in the model parameters as we measured successive responses; we

shall call this the information updating model. We shall now analyze the process whereby real changes in the model parameters occur as the student acquires new problem solving techniques. Knowledge of this process is essential for predicting how exposure to a given problem will affect the student. We shall rely on the learning model in the next chapter when we scan a set of problems to find the best example.

Prior to a learning event E our knowledge of the student's technique choice probabilities is represented by the Dirichlet distribution:

$$\{t|E\} = f_{\beta i}(y_1, y_2, \ldots, y_N | m_1, m_2, \ldots, m_N)$$

$$= \frac{1}{\beta(m_1, m_2, \ldots m_N)} y_1^{m_1} y_2^{m_2} \ldots y_N^{m_N} \quad , \; m_1, \ldots, m_N > 0$$

Since the posterior distribution for the event E is in turn the prior distribution for the succeeding event, it is convenient to require that the posterior distribution also have the Dirichlet form. Ideally we would like to "update" the prior distribution in the same manner as the information updating model. However the possibility of a real change in the student's technique choice probabilities creates a new uncertainty in our knowledge of the student. For instance, if we have updated our beta density prior distribution with 20 consecutive responses, assuming no learning, we have substantially reduced the variances

of the technique choice probabilities. However if the student subsequently is exposed to a better solution strategy, or encounters a new problem form in the same class, he may alter his response patterns over all the problems belonging to that state. It would then be incorrect to assume that our knowledge of his response probabilities is still represented by the simple Bayesian updating of the prior distribution.

Our state of information about the student can be imagined to progress monotonically between "discontinuities" that occur with each learning event. This is, of course, a strong assumption. The student's learning process and the observer's updating process, in general, are constantly competing. The discontinuity assumption was, however, suggested from observation of calculus students interacting with the methods of integration tutor. In Chapter 5 we show that calculus students do, in fact, exhibit response discontinuities. Qualitatively, a typical student episode went as follows:

1) the student encountered a new problem form
2) the student entered one or more failure states attempting a solution
3) the student received tutorial assistance
4) the student (often) worked several more problems before returning to #1 and #2.

It is clear that the distance between learning

discontinuities depends in part on the existence of "several more problems" in step 4 above. One can also argue that even if the student does not enter any failure states that learning is distributed among subsequent problems as the student experiences further successes with his new techniques. If this is true, the "discontinuity" is perhaps only a gradual change. However, Chapter 5 shows that for calculus students the change in response patterns is typically abrupt. This lends confidence to the assumptions that the learning process and the observing process can be considered independently and that learning typically occurs suddenly and sporadically.

What are the parameters of our distribution for the student after a learning discontinuity? Let us assume that the student is in problem state $s_i$, and chooses technique $T_j$, which results in an unsuccessful transformation. Although the student could choose another unsuccessful technique, let us assume that he then chooses technique $T_k$, which yields a succesful transformation. We shall label this sequence the event E. Since the student often asks for assistance when he enters a failure state, the choice of successful technique may be due to a tutorial hint. A first candidate for the student's technique choice probabilities after event E could be the prior distribution we are willing to use for a student before any contact with the tutor. A better candidate is this distribution modified by the new responses suggested to the student by the tutor after the student

entered the failure state. However this approximation does not reflect the student's previous work. Observation of calculus students suggests that the failing and successful techniques in event E are the techniques most substantially affected. We thus assume that the student's technique choice probability $t_{ij}$ (choosing technique $T_j$ from state $s_i$) that led from problem state $s_i$ to the failure state $s_{if}$ is modified by a learning parameter $\alpha$, and that the probability $t_{ik}$ of choosing technique $T_k$ (the subsequently successful choice) is modified by a related parameter $\eta$:

$$t_{ij,posterior} = \alpha\, t_{ij,prior} \quad , T(j) \text{ unsuccessful}$$

$$t_{ik,posterior} = \eta\, t_{ik,prior} \quad , T(k) \text{ tutor's choice}$$

$$\text{where } \eta = \frac{1 - \sum_{m \neq j,k} t_{im} - \alpha t_{ij}}{t_{ik}}$$

$\eta$ is a factor determined by $\alpha$ that renormalizes the sum $\sum_m t_{im,posterior}$ to 1. For instance, if $t_{i1}, t_{i2}, t_{i3} = 1/3$ are the only technique choice probabilities for state $s_i$, and $\alpha = 1/2$, then

$$\eta = \frac{1 - (1/3) - (1/2)(1/3)}{1/3} = \frac{3}{2}$$

If the failing technique $T_j$ is $T_1$ and the successful

technique $T_k$ is $T_2$, then

$$t_{i1,posterior} = \alpha(1/3) = \frac{1}{6}$$

$$t_{i2,posterior} = \eta(1/3) = \frac{1}{2}$$

$$t_{i3,posterior} = t_{i3,prior} = \frac{1}{3}$$

$\alpha$ is a factor depending on the problem and on the student. In Chapter 5 we show that in practice $\alpha$ can have a fairly wide range of values for a given student. We must also assume that $\alpha$ will change in time as the student becomes more experienced. In fact, it is clear that if the student is to eventually converge on the tutor's problem solving heuristics, $\alpha$ must approach 1 with increasing time. It is convenient to consider $\alpha$ itself as being distributed according to the beta function:

$$g(\alpha) = \frac{1}{\beta(r,s)} \alpha^{r-1}(1 - \alpha)^{s-1}$$

Now, however, the learning model product $\alpha t_1$ is no longer beta distributed. This is apparant from considering the simpler case where a probability p is beta distributed:

$$f(p) = \frac{1}{\beta(m_1, m_2)} p^{m_1 - 1}(1 - p)^{m_2 - 1}$$

and the product $\delta = \alpha p$ is distributed according to

$$f(\delta) = -\frac{d}{d\delta} \int_{\delta}^{1} \int_{\delta/p}^{1} f(p)\, g(\alpha)\, d\alpha\, dp$$

This integral is hard to evaluate in closed form, but if $\alpha$ is assumed to be uniformly distributed with $r = s = 1$, then

$$f(\delta) = -\frac{d}{d\delta} \int_{\delta}^{1} f(p)\, \alpha \Big|_{\delta/p}^{1} dp$$

$$= \int_{\delta}^{1} \frac{f(p)}{p}\, dp$$

If we also assume that p is uniformly distributed with $m_1 = m_2 = 1$, then

$$f(\delta) = \int_{\delta}^{1} \frac{1}{p}\, dp = -\ln(\delta)$$

which obviously is not a beta distribution.

Since it is useful to preserve the beta distribution form for the Bayesian updating procedure, we shall choose a beta distribution whose mean and variance are equal to the mean and variance of the product distribution[2]. In the simple case

discussed above,

$$\bar{\delta} = \overline{\alpha p} = \bar{\alpha}\bar{p}$$

$$= \frac{r}{r+s}\,\frac{m_1}{m_1+m_2}$$

and
$$v_\delta = \overline{\delta^2} - \bar{\delta}^2 = \overline{\alpha^2 p^2} - \bar{\alpha}^2\bar{p}^2$$

$$= \frac{r(r+1)}{(r+s)(r+s+1)}\,\frac{m_1(m_1+1)}{(m_1+m_2)(m_1+m_2+1)} - \frac{r^2}{(r+s)^2}\,\frac{m_1^2}{(m_1+m_2)^2}$$

If the new beta distribution has the form

$$f(\delta) = \frac{1}{\beta(n_1,n_2)}\,\delta^{n_1-1}(1-\delta)^{n_2-1}$$

then, equating means and variances,

$$\frac{n_1}{n_1+n_2} = \bar{\delta}\,,$$

$$\frac{n_1(n_1+1)}{(n_1+n_2)(n_1+n_2+1)} - \frac{n_1^2}{(n_1+n_2)^2} = v_\delta$$

---

2. This procedure is known as the method of moments.

and solving for $n_1$ and $n_2$ we get

$$n_1 = \frac{\bar{\delta}^2(1-\bar{\delta}) - \bar{\delta}\,{}^{v}\!\delta}{{}^{v}\!\delta}$$

$$n_2 = (1-\bar{\delta})\,\frac{\bar{\delta}^*(1-\bar{\delta}) - {}^{v}\!\delta}{{}^{v}\!\delta}$$

For the case discussed above with $m_1 = m_2 = r = s = 1$, $\bar{\delta} = 1/4$, ${}^{v}\!\delta = 7/144$ and we find that $n_1 = 5/7$, $n_2 = 15/7$. This provides us with the parameters for the beta distribution approximation of $-\ln(\delta)$. We then compare

$$\frac{1}{\beta(5/7,15/7)}\,\frac{(1-\delta)^{8/7}}{\delta^{2/7}} \cong 1.2903\,\frac{(1-\delta)^{8/7}}{\delta^{2/7}}$$

with $-\ln(\delta)$.

Figure 2.8 shows the two functions plotted. It is clear in this case that the Beta distribution is a very acceptable approximation to the exact product distribution.

The last few paragraphs have discussed a simplification of the model. When we generalize to a beta distribution involving both the unsuccessful and successful technique choice probabilities, the results are similar. In particular,

$$f(t|m) = \frac{1}{\beta(m_1,m_2,m_3)}\, t_1^{m_1-1}\, t_2^{m_2-1}\, (1-t_1-t_2)^{m_3-1}$$

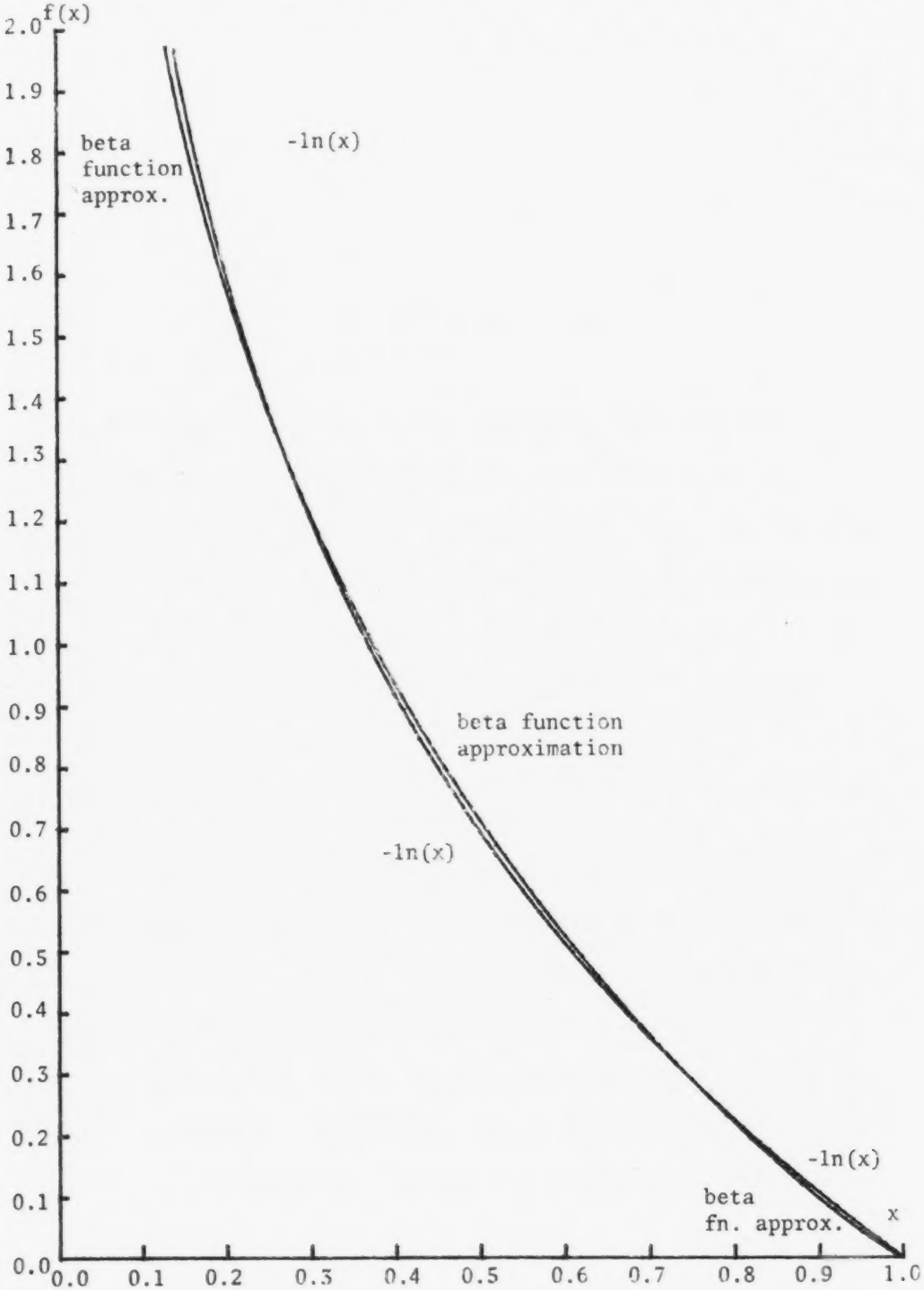


Figure 2.8. Beta distribution and the exact product distribution

where $T_1$ is the unsucessful technique whose probability is modified by $\alpha$, and $T_2$ is the successful technique whose probability is modified by $\eta$. The new beta distribution is

$$f(u|n) = \frac{1}{\beta(n_1,n_2,n_3)} u_1^{n_1-1} u_2^{n_2-1} (1-u_1-u_2)^{n_3-1}$$

where $u_1$ corresponds to $\alpha t_1$ ,

$u_2$ corresponds to $\eta t_2$ ,

and $u_3$ corresponds to $t_3$ .

Defining $R = r + s$ and $M = \sum_i m_i$ , we have

$$u_1 = \frac{r\ m_1}{R\ M}$$

$$v_{u_1} = \frac{(r + 1)\ r\ (m_1 + 1)\ m_1}{(R + 1)\ R\ (M + 1)\ M} - \frac{r^2\ m_1^2}{R^2\ M^2}$$

from the univariate analysis, and solving for $n_1$, $n_2$, and $n_3$ we have

$$n_1 = \frac{\bar{u}_1^2\ (1 - \bar{u}_1) - v_{u_1}\ \bar{u}_1}{v_{u_1}}$$

$$n_2 + n_3 = (1 - \bar{u}_1)\ \frac{\bar{u}_1\ (1 - \bar{u}_1) - v_{u_1}}{v_{u_1}}$$

$$n_3 = \frac{m_3}{M}\ (n_1 + n_2 + n_3)$$

These last five equations completely define the updating

process used by the tutor when the student encounters a learning discontinuity. In Chapter 5 we shall show that the instances of student learning discontinuities can be easily identified by the tutor.

We see now that the two processes complement each other. Starting with an original prior distribution of the student's technique choice probabilities, we update our student model until a learning event occurs (that is, until the student enters a failure state). At this point the learning model provides us with a new prior distribution, which we continue updating. The new prior distribution is linked to the old through the distribution of the learning parameter $\alpha$. Actually since the distribution of $\alpha$ will in general depend upon the student and upon time, we can imagine a third updating process involving $\alpha$ itself. As the student encounters successive learning events we the observers will improve our knowledge of $\alpha$. We could call this process "getting to know the student" For the time being we will assume that $\alpha$ has a fixed distribution independent of particular students. The challenge will be to deduce this distribution from an experimental environment

Expected Number of Steps

To obtain the student's transition probabilities, we could carry out a similar updating process for the technique result probabilities (the $q_{ijk}$'s), or alternatively, we can measure the

student's transitions directly. Once we have what we consider reasonable estimates for the $p_{ij}$'s we can calculate the student's expected number of steps to solution from any starting state. Using the theory of transient Markov processes, the matrix of expected delays before trapping, $[\tau]$, is related to the modified state transition probability matrix $P^*$ (created by removing all rows and columns of P corresponding to the trapping states) by

$$[\tau] = [I - P^*]^{-1}$$

where I is the identity matrix. Thus the sum $\sum_k \tau_{ik} = \nu_i$ is the sum of the delays in all possible states given that the system started in state $s_i$, and this is the expected number of steps to solution of a problem begun in state $s_i$. Equivalently, the product

$$[\tau] \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} \bar{\nu}_1 \\ \vdots \\ \bar{\nu}_N \end{bmatrix} = [\bar{\nu}]$$

is the column vector of the expected number of steps to solution from each state. Each of the terms $\tau_{ij}$ is useful as an indication of where the student is spending his time in the solution of a problem begun in state $s_i$. We can examine the expected posterior $\tau_{ij}$ for students with unusually long problem solutions to determine whether they are spending time in failure

states or in legitimate transformations.

Similarly it is helpful to know the variance of the delay in each state as well as the variance of the total number of steps. Again, from the theory of Markov processes, the matrix $\overset{v}{N}$ of variances of the expected delays, $\bar{\nu}_{ij}$, can be calculated by

$$\overset{v}{N} = \overline{N^2} - \bar{N} \square \bar{N}$$

where $\overline{N^2} = \tau[2(\tau \square I) - I]$

and $\bar{N} = \tau$, the matrix of delays.

Note that the box notation $A \square B$ represents the term by term multiplication operation for two matrices of similar dimensions, i.e. if $C = A \square B$, then $c_{ij} = a_{ij}\, b_{ij}$.

The column vector of variances of the total delays in the process is given by

$$\overset{v}{\nu} = (2\tau - I)\, \bar{\nu} - \bar{\nu} \square \bar{\nu}$$

where $\bar{\nu}$ is the column vector of the total expected delay in each state. Note that $\overset{v}{\nu} \neq \begin{bmatrix} \overset{v}{N} & 1 \\ 1 & 1 \end{bmatrix}$ because the times spent in each state are not independent random variables.

# Chapter 3

## The Tutorial Strategy

Chapter 2 introduced a general model of a student solving problems. From this model we defined important parameters of the student's understanding of the subject, such as his expected number of steps to solution, his probability of choosing a technique ($t_{ij}$), his probability of arriving in a given state after applying a technique ($q_{ijk}$), and the probabilities of entering or leaving a failure state ($p_{iF}$ and $p_{Fj}$). We discussed specific methods for encoding an observer's knowledge of the student and for modeling the student learning process. This chapter will present a real time tutorial strategy for computer assisted instruction that will use these models as its basis. The essential elements of the tutorial strategy are: student trouble thresholds which, when exceeded, cause the tutor to intervene in the student's problem solution; a set of problem solving priorities used by the tutor to give hints; two problem archives which the tutor can scan for problems that will optimally challenge or optimally help the student; and a self improvement scheme that allows the tutor to incorporate the best problem solving strategies of its students. In addition the tutor can both help the student apply techniques and can modify its own subject breadth to tutor students with differing backgrounds.

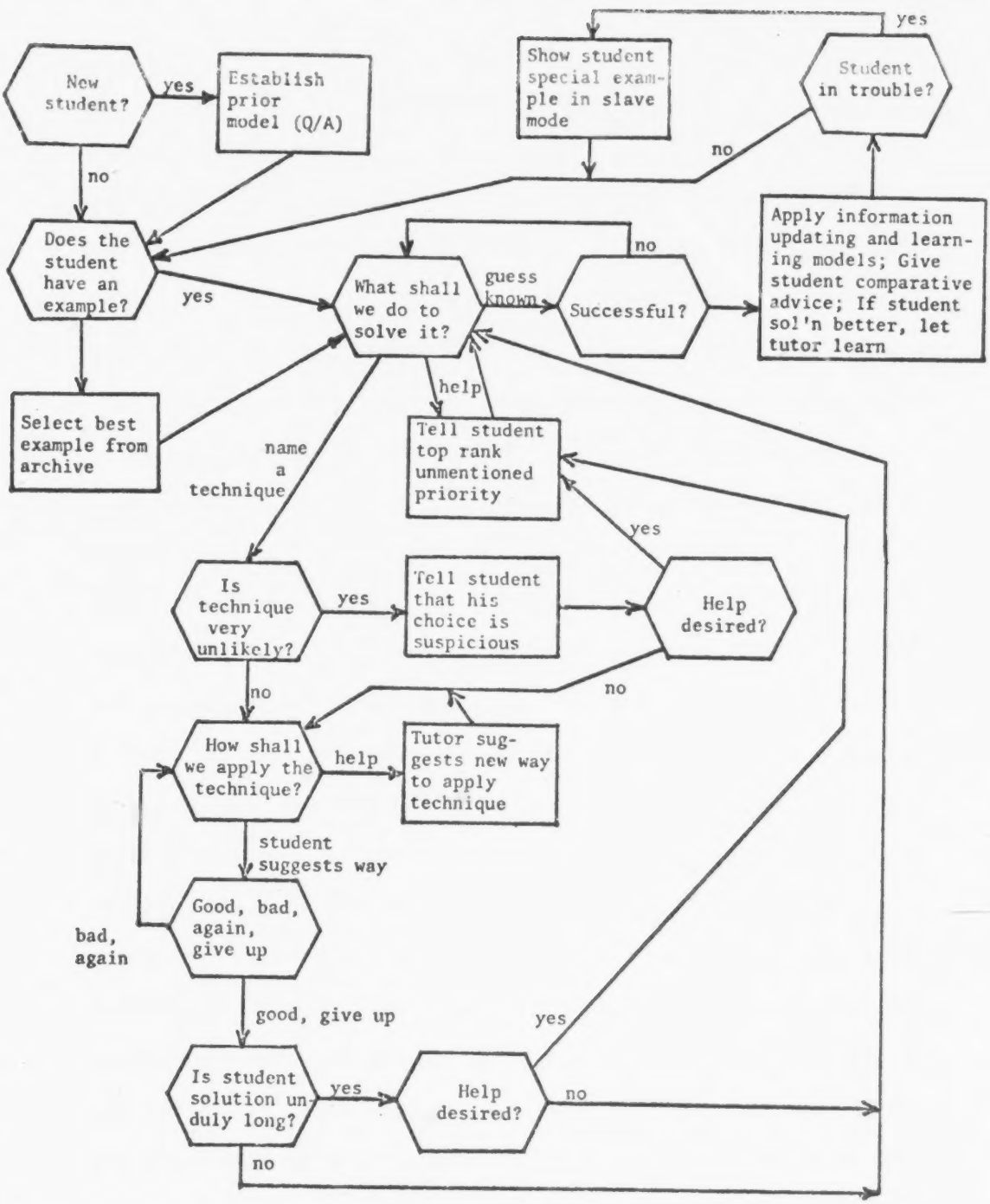


Figure 3.1. The flow chart of the tutor.

## The Plan of the Tutor

Figure 3.1 is the flow chart of the tutor. This chapter will discuss each element of the tutor in the order of execution of a typical tutorial episode. From a macroscopic viewpoint, the function of the four elements in the upper left corner of Figure 3.1 is to initialize the tutor's knowledge about the student and to select an example to work. The large circuit of ten elements in the center and lower portions of the figure handles the actual working of the example and the tutor-student dialogue. After the student successfully terminates the problem, the tutor performs certain bookkeeping functions and tests the student's problem solving patterns for signs of major trouble. This last phase is shown in the upper right corner of figure 3.1. The tutor then returns to find another example.

## Tutor Initialization: Dynamic Subject Scope

One of the characteristics of the tutorial phase of learning is the dissimilar subject backgrounds of the students. Often the students are involved in a lecture phase at the same time they are interacting with the tutor. Since the tutor is structured to deal with individuals it must be able to tune its level of presentation to the capabilities of each student. This is accomplished by querying each student as he logs in about which techniques he is familiar with. A new student would be asked if he knew each of the $N_T$ possible techniques. Thence afterward the

tutor would only ask him about those techniques he had not known in an earlier session.

During the session the tutor has the student's list of unknown techniques. Whenever the tutor encounters a situation where it would ordinarily give an unknown technique as a solution hint, it will warn the student that he may be venturing into deep water. The student then has the choice of aborting the problem solution as too difficult or choosing an alternative tutorial hint that he knows.

This process of dynamically altering the subject scope also allows the tutor to choose only problems from its archive that can be worked by techniques known to the student. Since an outline of the teacher's solution is stored with each archive problem, the tutor rejects any problem using unknown techniques. Thus the student in general has access to a subset of the problem archive. In fact we see that the concept of the dynamic subject scope generalizes the tutoring process since for a tutoring system with $N_T$ possible techniques, there exist $2^{N_T}$ possible subsubjects all tutorable by the same tutor.

## Selecting an Example: The Tutor-Student Distance

If the student has no example of his own, the tutor will select one from its example archive. In order to choose among its examples the tutor first calculates a "distance" measurement for each problem description state that expresses how much the tutor

and student disagree in choosing problem solving techniques. This distance is given by

$$D_i = \sum_j |T_{ij} - t_{ij}|$$

where $T_{ij}$ is the tutor's relative frequency of applying technique $T_j$ in state $s_i$ and $t_{ij}$ is the student's expected relative frequency of applying the same technique. The tutor's $T_{ij}$ is determined by scanning all of the problems in the problem archive, searching for occurrences of state $s_i$. The tutor-student distance can range in value from 0 to 2 for each state, corresponding to complete agreement or complete disagreement, respectively. The following paragraphs discuss alternative schemes for choosing the best problem for the student using the tutor-student distances $D_1,\ldots,D_N$.

If we know in advance what responses the student would make, we could select the problem that would minimize the total distance

$$D_T = \sum_i D_i$$

posterior to working the problem. Lacking this perfect information, we could nevertheless calculate the probability of the student choosing technique $T_j$ in state $s_i$, given his response probabilities $t_{ij}$. For instance, if the original problem description state is $s_k$, this probability would be

$$P\{T_j|s_i\} = \delta_{ik}\, t_{ij} + p_{ki}\, t_{ij} + \sum_{l} p_{kl}\, p_{li}\, t_{ij} + \sum_{l,m} p_{kl}\, p_{lm}\, p_{mi}\, t_{ij} + \ldots$$

where $p_{rs}$ is the probability of the student making a transition from state $s_r$ to state $s_s$.

The change in the student's total distance $D_T$ could then be estimated by carrying out this calculation on each candidate problem for all possible states and techniques.

Apart from the computational complexity of this scheme, there are two significant objections to its use. First, we would find that all problems of the same initial problem description state give identical predicted contributions to the change in $D_T$. This still leaves us with a choice to make among a possibly large number of problems. The second objection is that such a computation ignores the solution used by the tutor for the particular problem except insofar as it contributes to the tutor's $T_{ij}$. What is needed is a model to predict what relation the student's particular solution will have to the tutor's particular solution.

To pose such a model, we assume that if the tutor-student distance for a given state is large, the student is more likely to

enter failure states and more likely to get into situations where the tutor gives him a hint that may change his problem solving patterns. Although the tutor's hints are not based on its own particular solution, the tutor does compare its solution with that of the student upon completion of the problem.

A reasonable measure of a candidate problem is the set $\{D_i, D_j, D_k, \ldots, D_m\}$ of tutor-student distances for states encountered in the tutor's solution. The expected example distance $D_E$ of this problem is the weighted sum

$$D_E = D_i + p_{ij} D_j + p_{ij} p_{jk} D_k + \cdots + p_{ij} \cdots p_{nm} D_m$$

where the transition probabilities are those of the student. The expected example distance has the following desirable features: 1) it is computationally tractable; 2) its value is proportional to the expected occurrence of tutor hints and comparisons that will change the student's $t_{ij}$'s; 3) it depends on the entire tutor solution and will yield very few ties among candidate problems; and 4) since it is weighted by the student's transition probabilities it takes into account the possibility that the student may diverge from the tutor's solution.

To select an example, the tutor calculates the expected example distance for all the unworked archive problems, eliminating those using techniques unknown to the student, and chooses that problem with maximum $D_E$.

## Choosing a Technique

When the example problem is established, either by student initiation or archive selection, the student is presented with the fundamental question: "What shall we do to solve it?". He then has three fundamental choices. He may name one of a set of problem transforming techiques; he can attempt to finish the problem directly by either guessing the final answer or correctly identifying the integral as "known"; or he may ask the tutor for a hint. The following paragraphs discuss how the tutor handles each of these options.

## Unusual Technique Threshold

If the student decides to name a problem transforming technique, the tutor needs to measure the appropriateness of the response. In keeping with the goal of giving the student as much freedom as possible, the tutor should not comment on the student's choice of technique unless the tutor thinks the student made a very poor choice.

We define a simple threshold that causes the tutor to intervene whenever the student chooses a technique that is unlikely, in the tutor's view, to provide a successful transformation in comparison to other untried techniques. If $\epsilon_i$ is an adjustable quantity between 0 and 1, depending upon the state $s_i$, which we call the unusual technique threshold parameter, then we define the unusual technique threshold $Th_{ti}$ as:

$$Th_{ti} = \epsilon_i \left[\max_j \; t_{ij,tutor}\right]$$

In other words, the threshold is a certain fraction of the tutor's response frequency for the most likely technique in that state. The unusual technique threshold is exceeded whenever technique $T_j$ chosen by the student satisfies

$$t_{ij,tutor} < Th_{ti}$$

In other words, the tutor looks at its own priorities to decide if the student chose a technique below the tutor's relative frequency threshold.

If the tutor's technique probability falls below the threshold, the tutor will stop the student to ask if he would like a hint since his choice is suspicious. If the student desires to proceed with his "unlikely" technique, he must be allowed to do so, since it is possible that he is pursuing a line of reasoning that is not represented in the tutor's archive. If the student opts for a hint, it is given to him and he returns to "What shall we do to solve it?".

Notice that if the student is in a failure state where he has tried one or more techniques unsuccessfully already, he will not necessarily be more likely to cause tutor intervention. Although the tutor's archive contains no occurrences of failure states, the tutor knows what its adjusted priorities would be if this most likely technique failed. The probability for the most

likely technique is demoted by the failure parameter $\theta$, and all the rest are increased for the sake of normalization. This procedure insures that the student will be left alone by the tutor unless he tries something quite unusual.

## Hint Generation

When the student asks the tutor for help in choosing a technique, the tutor must respond from its own knowledge of how to solve the problem. The crucial point is that the tutor does not know how to solve the problem! If a tutor is to respond to arbitrary student problems and solution paths, the tutor cannot store certain prescribed solutions in its memory. In fact, many subjects allow two or more solution paths for most of their problems. The tutor cannot use the particular solution stored in its archive since usually the student either suggests his own problem or deviates from the solution path used by the tutor. The tutor derives its own response frequencies from the archive by ranking the frequency of the various techniques applied for each problem type. We refer to this ranking as the tutor's priorities. When the student asks for help, the tutor suggests the highest priority technique. Successive requests for help yield successively lower priority technique choices. We can thus state the principle:

The tutor provides technique choice advice by presenting

the student with its own technique choice priorities.

It is possible that the most highly recommended technique will not solve the problem. The student should be prepared to fail occasionally even with "good" advice and start the problem over using the next most highly recommended technique. Since the student is learning a process of problem solving, rather than the solutions to isolated problems, even such negative experiences will broaden his judgement by causing him to search for less likely solution schemes.

## Applying the Technique

If the student avoids triggering the unusual technique threshold, he enters a subprogram specifically designed for the technique. He now is exposed to the second level version of "What shall we do to solve it?". In this case the student can suggest a solution scheme (such as "let u = x" in a substitution, or "let $u = e^x$, dv = sin(x) dx" in integration by parts, "apply the half angle identity" in trigonometric identities. Alternatively, the student can ask for help. Following the application of the technique, the student has a chance to view the result and accept it, reject the result and try again, accept the result and apply the technique again, or give up on the technique altogether.

Theoretically, we could generate a set of priorities for the student when he wants help with applying a technique as we do

when he wants help with choosing a technique. Eventually, however, we must stop naming techniques and subtechniques and actually take the student through a manipulation from start to finish. This is a practical rather than theoretical choice. It is possible to imagine an optimization scheme for applying a technique that would involve searching all the paths that could be generated by different applications of the technique and then choosing the path that led to the state with the lowest expected number of steps to solution. The objection to this procedure is the unreasonable overhead that would result from this real-time decision. In the methods of integration tutor described in Chapter 4, the explanations of technique applications are handled by specific algorithms tailored in each case to the technique. Of course, these algorithms contain procedures for rejecting problems unsuited to the technique. These kinds of predetermined decisions are termed "wired in heuristics". It is important to choose the state definitions for any tutorial system so as to diminish the importance of wired in heuristics. In particular, any decision point that allows a genuine divergence of opinion among reasonable problem solvers must not be handled by an algorithm that always chooses one type of solution. If such a situation arises in the construction of the tutor, a separate state should be constructed that allows the student to choose among several paths and which allows the tutor to apply the techniques just developed.

Problem Length Threshold

Following the application of the technique, the tutor must know if the problem solution is getting unusually long. It is often possible to perform a very large number of steps on a simple problem without triggering the unusual technique threshold. From Chapter 2, however, if we know the tutor's state transition probabilities (the $p_{ij}$'s) we can calculate the expected variance of the number of transitions to stop, given that the problem started in state $s_i$. From this we can establish a problem length threshold :

$$Th_{1i} = \bar{\nu}_i + \kappa\sqrt{\overset{v}{\nu}_i}$$

where $\bar{\nu}_i$ is the expected number of steps starting from state $s_i$ (the mean delay from state $s_i$ to a trapping state), $\overset{v}{\nu}_i$ is the expected variance of the delay from state $s_i$, and $\kappa$ is a number we call the problem length threshold parameter. The tutor interrupts the student whenever his solution length exceeds the number $Th_{1i}$, defined as the tutor's mean number of steps plus $\kappa$ standard deviations.

When the student exceeds the problem length threshold, the tutor will intervene to ask if the student wants a hint. The tutor can not in general know exactly why the student is producing such a long solution, and of course must not force the student to terminate his solution. However, practice suggests that returning

to the beginning of the problem and reexamining the technique priorities will usually prompt the student into a better solution. Students who produce unwieldy solutions to ostensibly short problems usually have not asked the tutor for suggestions.

## Finishing the Solution

The student can continue to apply techniques to a problem indefinitely. Each such application involves one loop of the lower central portion of the flowchart in Figure 3.1, returning each time to "What shall we do to solve it?". Eventually the student will reduce the problem to a simple, recognizable form. If this form is one of a list of agreed upon "known" forms, the student can simply type "KNOWN" to terminate the problem. The student may also try to guess the final answer, even if the problem is not of the known form. Finally, if the student must stop working on the problem before it is normally solved, he may give up.

Following completion of the problem, the tutor updates its prior estimates of the student's $t_{ij}$'s by using the information updating and student learning models described in Chapter 2.

The tutor then prints a summary of the techniques employed by the student to solve the problem. If the problem came from the archive, the tutor also prints its own solution alongside the student's. This is a very effective way for the student to compare his problem solving schemes with those of the tutor,

particularly if he has solved the problem without tripping the tutor intervention thresholds or without asking for a hint.

## Tutor Learning

Much of the value of the tutoring process we have developed in this chapter depends on the tutor being a good problem solver itself. In particular, the technique choice convergence schemes we proposed for problem selection would be counterproductive if the student was a better problem solver than the tutor. In this case the tutor would be attempting to bring the student down to its own level. The use of this tutorial scheme would also be severely restricted if the tutor required initialization by some kind of grand master of the subject. Therefore a most important development in our tutorial theory is a self improvement strategy for the tutor. We want the tutor to recognize superior student solutions and learn them in such a way that all future tutoring decisions will reflect the new knowledge.

The tutorial system as we have described it thus far is well suited for modification of the tutor's strategies. Except for the wired in heuristics all tutorial responses are determined by the tutor's technique choice probabilities and the two problem archives. From a practical standpoint these can be considered as volatile as any other piece of data.

The real problem is to identify a criterion for superior student solutions. In particular the tutor cannot recognize

brilliance in the solution of a problem that does not exist in its own problem archive. We must remember that the tutor's technique choice probabilities are determined entirely from the problem solutions in the general problem archive. A new problem can only be judged as some statistical combination of the tutor's previously known problems, thus the measure of its true difficulty is unknown.

We shall make the simple assumption that length of problem solution is a measure of superiority. Thus whenever the student works a problem from the general problem archive that is shorter than the tutor's solution the tutor will remember the student's solution by replacing its archive entry and updating its $t_{ij}$ matrix (by subtracting the old solution statistics and adding the new). The tutor must of course reject solutions that end in the "give up" trapping state or involve the "guessing" technique if such a technique is allowable. In this way the tutor's basis for heuristic decisions can eventually be altered by the students.

Other superiority schemes that would not necessarily shorten the problem solution can be easily imagined. For instance if technique $T_i$ is thought to be more elegant than $T_j$ then any solution using $T_i$ could replace ones involving $T_j$, possibly subject to constraints on the total length of solution. Going one step further, the problem archive could be optimized at several levels simultaneously, depending upon different expected subject scopes. In other words the archive could have several disjoint

levels, each level depending on how many techniques the student knows.

Diagnosing poor problem solving practices

Because we have committed the tutor to allow the student exceptional freedom while solving problems, it is quite possible that a confused student may solve whole classes of problems poorly without receiving much warning from the tutor. Thus after each problem, we ask the tutor to scan the student's overall problem solving patterns for signs of trouble.

During the course of solving a problem, the tutor interrupted the student in the act of choosing a technique if the tutor felt that the student's choice was unusual enough to be reconsidered in favor of the tutor's. Similarly if the tutor-student distance $D_i$ for the $i^{th}$ state is sufficiently high, the tutor will stop the tutoring session to show the student a complete example. The tutor assumes that the student's problem solving techniques at this point are sufficiently bad to require that the student solve the problem in a "slave" mode that only allows him to proceed with the tutor's recommended solution. We define a problem solving trouble threshold for each state by:

$$D_i > T_{p_i}$$

where $T_{p_i}$ is a parameter, depending upon the state $s_i$, chosen

between the extreme possible values of 0 and 2.0 For instance, we may decide that one particular state is more critical than others and thus assign it a lower problem solving trouble threshold.

When the student exceeds the problem solving trouble threshold the tutor then scans a special archive of problems reserved for this situation. Each problem in the archive is stored with the complete set of responses that the teacher used to work the problem. The student then begins the problem as usual but is stopped every time he does not agree with the teacher's response. Needless to say, the problem solving trouble thresholds should be set high enough that this procedure is invoked relatively rarely since it is a brute force effort to move the student's technique choices toward the tutor's. In practice with the methods of integration tutor a threshold value of 1.75 was found to be reasonable.

The advantage of entering the "slave" mode in this tutorial situation is that we can be sure for the purposes of optimum problem choosing that the student will see all the steps. This was the assumption we could not make when we chose example problems from the general archive. We also tacitly assume that the effect of a forced response is the same as a voluntary response in the non-slave mode. The steps for choosing an optimum problem for the slave mode are:

1) stop the student when he exceeds the problem solving trouble threshold $Tp_i$
2) use the information updating model to calculate the student's expected posterior technique choice probabilities as a result of being exposed to the entire problem.
3) calculate the tutor-student distance $D_i$ with the student's new $t_{ij}$'s.
4) Minimize the value of step 3 over all the problems in the archive.

After the student is shown the problem, we continue with the prior distribution that the learning model predicts for the student after the complete exposure to the new techniques.

Summary

This chapter has presented a computer tutorial system applicable to a wide variety of subjects and capable of providing the student tutorial assistance at several levels. The tutor bases its own problem solving heuristics entirely upon a general problem archive established by the original human teacher. Using the problem archive the tutor can select optimal problems either as examples to set a floundering student on the right path or as problems designed to challenge the stronger student in troublesome areas. Because all of the tutor's recommendations are derived by

statistically averaging over the entire archive, the student can initiate his own problem and expect to receive the same level of tutoring that he would get if the tutor chose a problem from the archive. At the technique choice level the tutor offers its own technique choice preferences whenever the student asks for help or exceeds the unusual technique threshold. At the technique application level the tutor relies on wired in heuristics to make specific suggestions but leaves the final decision of application up to the student if a choice exists.

In addition to the main function of providing tutorial assistance the tutor also dynamically alters its subject scope for the individual student and can optimize its own performance with respect to any measurable superiority criterion.

# Chapter 4

## The Methods of Integration Experiment

This chapter describes an experimental "methods of integration" tutor developed from the ideas of the last two chapters. We shall show how the states and techniques for this subject were defined and how the tutor thresholds and problem archives were implemented. A discussion of the student information updating process and two versions of the student learning model will follow. Finally we describe in a general way the challenges of creating computer programs that tutor this subject. A discussion of the results of the experimentation with calculus students is deferred to Chapter 5.

### The Choice of Methods of Integration

Methods of integration was a good choice of subject for this tutorial system for many reasons. As a subject in a calculus course, it is almost never taught as an algorithmic procedure (like differentiation). Rather the emphasis is on the acquisition of a number of techniques like substitution, integration by parts, and partial fraction expansion. Although the student is often given groups of problems solvable by the same techniques, the real challenge is the recognition of the correct approach, rather than the details of the technique application. In addition, most problems can be solved by any one of several approaches involving

different technique choices and different lengths of solution, thus providing the tutor a complete range of possible student results to judge. This unusually rich problem solving structure is ideal for testing the generality of the tutorial methods proposed in Chapter 3. At the same time, the tutorial strategy is not designed solely for this subject, as the examples in Chapter 3 point out.

## Identification of the States and Techniques

In the initial phases of development of the integration tutor it was hoped that the state definitions could be kept completely independent of problem solving considerations. The goal was to have each state unambiguously defined so that the tutor could know which state the student was in. Although this remains as an ideal, it was found that in certain situations a state definition dependent upon "the way the problem is solved" is preferable to the pure problem description approach.

For instance, in the case of problems involving simple variable substitutions leading directly to "known" integrals, integral solvers overwhelmingly recognize these problems as a distinct class based upon the substitution approach. Although one could define this class using exclusively structural properties (the presence of a term and its derivative,...) the motivation for doing so is still based on the way the student solves this class. The key point is that virtually every integral solver solves these

problems with a simple variable substitution, and it is unrealistic for the tutor to lump these problems into other classes that could yield a variety of possible tutoring hints. Thus as an exception to our rule for defining states (in Chapter 3), we list as state #2 below the state of "recognized substitutions". Similarly, state #3 is also an exceptional state: the state of "recognized trigonometric substitutions". Except for the trapping states, all of the remaining states are defined by their problem descriptions; the table of problem state names follows:

0. The Solved state
1. Known integrals
2. Recognized substitutions
3. Recognized trigonometric substitutions
4. Trigonometric & hyperbolic functions
5. Exponential functions
6. Arc-trigonometric and -hyperbolic functions
7. Fractional powers of functions
8. Combination of types 4 & 5
9. Combination of types 4 & 7
10. Combination of types 5 & 6
11. Combination of types 5 & 7
12. Combination of types 6 & 7
13. Polynomial functions
14. Other
15. The Give-up state

Problem state 1 is defined as the set of those integrals agreed upon by the student and tutor as requiring no more transforming to reach a solution. These integrals are sometimes solved in the Lecture phase of the student's learning by calculating the limit of an infinite sum, but are rarely solved by

the student after that point. These integrals are given to the student at the beginning of the tutoring session.

In each of the types 4 through 12 above the characteristic functions identifying the state can be multiplied or divided freely with polynomials in the variable of integration. Thus

$\int X^2\,dX$ is classified in state 1,
(but not $\int (3*X)^2\,dX$ )

$\int \sin(X)*e^{\cos(X)}\,dX$ is classified in state 2,

$\int \frac{1}{X^2 + 5}\,dX$ is classified in state 3,

$\int (X^3 + X)*\sin(X)\,dX$ is classified in state 4,

and $\int \frac{X^2 + 2*X + 5}{X^3 + 17*X^2 - 1}\,dX$ is classified in state 13.

In addition we will define one student failure state for each problem state given above (other than states 0 and 15), giving us a total of 30 states. State 0 is achieved when the student successfully identifies a known integral. State 15 is achieved only when the student requests to give up on the problem. As explained in Chapter 2, these two special states are the trapping states of the process. Note that no state corresponding to a combination of types 4 and 6 is given since integrands involving both trigonometric and arc-trigonometric functions are

virtually nonexistent. Such a problem, if encountered, would be classified in the "other" state, #14.

In a like manner we list the techniques of transformation that we allow the student to apply to integral problems:

1. The Known integral routine
2. Ordinary substitution
3. Integration by parts
4. Trigonometric substitution
5. Trigonometric identities
6. Separation of the sum
7. Polynomial division
8. Completion of the square
9. Partial fraction expansion
10. Conjugation of the denominator
11. Expansion of a power
12. Returning to the previous integral
13. Guessing the answer
14. Giving up

## The Tutor as Seen by the Student

Logging in. When the student logs in to the tutor for the first time, the tutor must establish the scope of the student's understanding of the subject. The tutor asks the student seven yes-or-no questions:

1. Have you ever studied integration by parts?
2. Have you ever studied trigonometric substitution?
3. Have you ever studied trigonometric identities?
4. Have you ever studied polynomial division?
5. Have you ever studied completion of the square?
6. Have you ever studied partial fraction expansion?
7. Have you ever studied conjugation of the denominator?

Several techniques were assumed known by the student, such

as simple substitution, separating a sum, expansion of a power, applying the known integral routine, and guessing the answer. All but the first of these are simple logical or algebraic manipulations that would be prerequisites for any exposure to methods of integration. Simple substitution was not included in the list because nearly all students learn this technique first. If a student logged in who proclaimed ignorance of every technique including simple substitution, the tutor would not be able to give intelligible hints for any nontrivial group of problems. In other words, it is assumed that any student who works with the tutor is at least aware of the technique of simple substitution.

Since the student usually will increase his repertoire of problem solving techniques during the period he is interacting with the tutor (perhaps by outside reading or Lecture phase exposure), each time the student logs in, the tutor asks him those questions to which he responded negatively in the past. The tutor keeps a statistical summary file on each student, one item of which is the monotonically decreasing list of "unknown" techniques.

Choosing the problem. After the student has logged in he is asked "Do you have a problem?". If the student has a problem he responds with "yes" and then types in his integral. If the student responds "no" the tutor then retrieves the statistical summary file for the student and constructs an appropriate prior

set of statistics. This process is explained in detail in a later section. Armed with the prior statistics the tutor scans the problem archive file, calculating the estimated weighted distance between the tutor and student for each problem as described in Chapter 3.

The problem that yields the highest value is then chosen for the student. In practice, the problem selection process takes approximately 3 seconds of machine time, a not unreasonable delay for the student.

Choosing the technique. After the problem is selected the student must choose from among his repertoire of techniques. The tutor types the integral and follows with "What shall we do to solve it?". The student refers to a printed list of abbreviations for the 14 techniques listed above. He may specify directly any of the techniques or he may type "HELP" or "REVIEW". HELP causes the tutor to show the student the name of the technique the tutor thinks is most likely to solve the integral. Successive HELPs give successively less likely hints until the tutor's hints are exhausted. REVIEW causes the tutor to show the student all the steps he has performed so far, in case he has made a number of transformations and is confused as to the status of the problem.

Applying the technique chosen. Once the student has chosen the technique he is placed under the control of the program

specific to that technique. If the technique involves a secondary choice, such as in substitution, the tutor types: "Can you think of a substitution?". The student then can answer "YES", "HELP", or "EXIT". If he types "YES" he then proceeds to type in the actual expression for which he is substituting. If he types "HELP", the wired-in heuristics of the program take over to find a reasonable substitution. In this case the program gives the student a choice among several conceivable candidates. For instance, with the integral

$$\int x * e^{x^2}\, dx$$

either the substitution $U = x^2$ or $U = e^{x^2}$ will yield a known integral. The tutor will present one of these to the student and ask him if the choice is reasonable or whether he would like another candidate or whether he would like to give up on substitution altogether. Thus the final decision of which substitution to make is left up to the student even though the tutor's own wired-in heuristics did the original work. Usually the students treated the search for a technique application choice as a challenge, prefering to use the HELP feature as a last resort.

If the technique involves no choice of action, as in polynomial division, the tutor simply prints the answer out and returns to "What shall we do to solve it?".

A Sample Student-Tutor Dialogue

In order to capture the flavor of the interaction between the student and the tutor, we present a sample dialogue selected from actual protocols gathered during an experiment with calculus students. In the following, the underlined characters are those typed by the student, everything else is typed by the tutor. Statements beginning with a semicolon (;) are explanatory comments inserted by the author. They did not appear to the student.

(INTEGRATE)

; Student's command to LISP to start the tutoring program.

Hello! What is your name?

Ruth

Hello again, Ruth

Have you ever studied partial fraction expansion?

no

; The tutor recognizes the student Ruth and asks if
; she has learned partial fraction expansion yet.
; She had previously answered "yes" to all the other
; techniques.

Ok, do you have a problem?

no

All right, I will select one for you!

Thinking for a moment...

; At this point the tutor retrieves Ruth's technique
; choice probability matrix (the $t_{ij}$'s) from an
; external file, constructs an appropriate prior,
; calculates the expected example distance for each
; problem in the example archive, and chooses the problem

; with the highest resulting value.

Ok, Ruth

Our integral is $\int X*\log(X)\ dX$

What shall we do to solve it? : sub

;Ruth has chosen substitution

Can you think of a substitution? (YES, HELP or EXIT)

; We are now in the substitution program and Ruth has
; 3 choices: 1) specify her own choice of substitution
; 2) receive a hint from the tutor
; 3) abort substitution altogether and
; return to the original integral

yes

Let U1 = log(X)

; The tutor prompts the student with "Let U1 = " and
; she responds with her choice. Successive substitutions
; would involve the variables U2,U3,...

Ok, Ruth

we will let U1

equal log(X)

; At this point the tutor calculates dU = 1/X dX
; and tries to find dU in the integrand.
; Since the search was unsuccessful, the tutor then
; solves U1 = log(X) for X so that the substitution can
; be carried out.

Thus X

equals $e^{U1}$

and dX

equals $e^{U1}$ dU1

which we substituted into the integrand

Our new integral is now $\int U1*e^{(2*U1)}\ dU1$

Please type Good, Bad or Giveup

; Now that the student has applied substitution,
; the tutor lets her choose whether to go on with
; this new integral. "Good" will cause this new integral
; to become the "official" new transformation, "Bad"
; will return us to the start of the substitution routine
; for another try, and "Giveup" will abort substitution
; to return to the original integral.

bad

Can you think of a substitution? (Yes, Help or Exit)

; We are now back to the original integral

help

I can't find a suitable substitution! Will exit...

; The tutor could not find a substitution that it felt
; led to a simpler integral. The point, of course, is that
; substitution was not a very good choice for this
; problem.
; Notice however that the tutor intervention threshold was
; not tripped when the student specified substitution. The
; intervention parameter is deliberately set high enough
; so that the student has some latitude to try poor
; guesses and see the results.

Ok, Ruth

Our integral is $\int X*\log(X)\ dX$

What shall we do to solve it? : help

; Now the student wants a hint from the tutor

I suggest you try integration by parts

What shall we do to solve it? : parts

; The student has taken the hint and asks for
; integration by parts

Integration by parts!

We will use the notation

$$\int U\,dV = U*V - \int V\,dU$$

Can you think of a choice for U ? (Yes, Help or Exit)

; The student has essentially the same choice as she did
; upon entering the substitution program

help

How about letting U = log(X)

Can you think of a choice for U?

yes

Let U = log(X)

; Apparantly the student is somewhat unfamiliar with
; integration by parts, since she did not even attempt
; to guess a choice for U.

Ok, Ruth

we will let U = log(X)

thus dV = X dX

hmmmm...

; A slight pause as the computer utilizes a simple set of
; wired in heuristics to find the integral of dV. If the
; tutor cannot solve the integral of dV by the known
; integral routine, simple substitution, or simple
; trigonometric substitution, the student is so informed
; and the tutor returns to asking for a choice of U.

An intriguing choice!

$$\text{thus } V = \frac{X^2}{2}$$

$$\text{and } dU = \frac{1}{X}$$

We now have $\int X*\log(X)\ dX$

$$= \frac{X^2 * \log(X)}{2}$$

$$-\int \frac{X}{2}\, dX$$

Please type Good, Bad, Again, Solve or Giveup

; Good allows the student to exit with the new integral
; Bad recycles back to the beginning of integration by
; parts for a new choice of U
; Again lets the student integrate the new integral by
; parts again (not used in this sample protocol)
; Solve lets the student solve for the original integral
; if at least two consecutive integrations by parts
; have been done (not used in this sample protocol)
; Giveup aborts integration by parts and returns with the
; original integral

Good

Ok, Ruth

Our integral is now $\int X \; dX$

times $\frac{-1}{2}$

; Extraneous coefficients are always factored out of the
; integrand

What shall we do to solve it? : known

; The students often used "guess" at this point to
; guess the answer to simple integrals. "Known" however
; simply provides the answer once the known integral is
; successfully identified.
; Notice that the student has now completed the problem.
; If this problem had dragged on one more step, the tutor's
; problem length threshold would have been exceeded and
; the tutor would have taken the student back to the
; beginning of the problem to find a better solution.

Very Good!

Before resubstitution

the answer to integral 1

$$\text{is } \frac{-X^2}{4}$$

Adding the term from integration by parts

$$\text{we have } \frac{2*X^2*\log(X) - X^2}{4}$$

plus an arbitrary constant...

You did that problem in 3 steps compared to 2 for me!

| My Solution | Your Solution |
|---|---|
| 1. Integration by parts | 1. Substitution |
| 2. The known integral routine | 2. Integration by parts |
| | 3. The known integral routine |

; One of the most effective features of the tutor is this
; problem comparison. Even though the student does not
; see the exact step-by-step solution of the tutor, the
; comparison of the solution outline is usually revealing.

Thinking for a moment.....

; The tutor now adds the pattern of responses generated by
; this problem to the statistical summary for Ruth. A
; trouble value is calculated for each problem state (See
; Chapter 3). If the trouble value for any state
; exceeds the problem intervention threshold, the tutor
; selects an example from a special list of remedial
; problems -- and forces the student through the entire
; solution. This did not happen in the protocol presented
; here.

Ok, do you have a problem?

; We have now come back to the starting point shown above.

## Implementation of the Thresholds and Archives

Summarizing the results of Chapter 3, we defined three

thresholds that affect the dynamic performance of the tutor. The technique intervention threshold and the problem length threshold were used by the tutor in the course of a student problem solution to challenge an unusual choice of technique or an unusual length of solution. The problem intervention threshold was used at the end of a problem solution to see whether the student was developing critical trouble in one or more problem states. If the threshold was exceeded, the tutor did not allow the student to select the next problem, but forced him to look at a specially chosen example.

As is explained in Chapter 5, the tutor evolved in three stages. Stages 1 and 2 were followed by experimentation with students learning calculus. Stage 3 was followed by the present report. Unfortunately, although virtually all the other salient features of the tutor existed in some form by stage 1, the technique intervention threshold and the problem length threshold were installed during stage 3 and did not undergo a thorough evaluation by real calculus students. Preliminary results with the threshold settings described in this chapter will, however, be presented in Chapter 5.

Chapter 3 introduced the unusual technique parameter $\epsilon$ to define the unusual technique threshold. In practice, we have used a value of $\epsilon = 0.25$ with success. Thus if in a given state (failure states included) the tutor's most likely technique choice $T_m$ has probability $t_{im}$, then the threshold is exceeded whenever

the student chooses a technique $T_j$ with tutor-probability $t_{ij} < 0.25\ t_{im}$ .

Similarly for the problem length threshold we chose a value of 2.0 for the problem length parameter. This means that the student is stopped for a review whenever his problem solution runs more than 2.0 standard deviations longer than the mean of the number of steps for problems of this class (tutor's statistics).

The problem solving trouble threshold was set at 0.5 for problem states #2 and #3 (simple substitutions and simple trigonometric substitutions) and 1.75 for the other states. This had the effect of concentrating the tutor's attention on these two states since the student could not make more than one or two failures in these states before the tutor-student distance exceeded 0.5. The lessons learned through interaction with the calculus students are discussed in the next chapter.

The general problem archive consists only of the original problem description (the integrand) and a list of ordered pairs of the form

(state,technique) , (state,technique), ...

representing the tutor's own solution of the problem. Note that a complete reconstruction of the tutor's solution is not possible because information on <u>how</u> the tutor applied the techniques is not given. All that is known is which states the tutor arrived in, and which techniques it subsequently employed.

This form is sufficient to store all the information needed to scan the archive for an optimal problem (as described in Chapter 3), and to compare the length of the student's solution with that of the tutor. If we also stored each of the specific responses needed to work the problem in detail (necessary only for slave mode problem selection) we could combine the two archives into one and the students could conceivably improve every problem the tutor could give them. This was not done simply because the general problem archive would have tripled in length, resulting in increased overhead each time it was scanned. In addition, the detailed response information is not used except when the student is in the slave mode.

Whenever a student works a general archive problem in fewer steps than the tutor, the tutor automatically rewrites the general problem archive with the student's solution outline replacing the teacher's. The tutor also rewrites its own record of technique choices from which its technique choice probabilities are calculated for every student problem. Thus every student on the system is exposed to the new tutorial strategy immediately after the solution to an archive problem is improved. Since the general archive contained about 80 problems, the effects are not dramatic each time a problem is rewritten, but the cumulative effect is substantial. Notice that certain precautions must be taken to screen trivially improved solutions from supplanting those of the tutor. One of the transformation techniques is

"guessing the answer", a popular choice by students confronted with integrals from class 2 or class 3, such as

$$\int \frac{1}{X + 3}\, dX = \log(X + 3) \quad .$$

In this case the tutor will assume that the student knew that the expected method of solution was simple substitution and will treat the solution record as if simple substitution had been used. On the other hand, the student can sometimes successfully guess the answer to more complicated integrals for which no canonical solution can be assumed. A sufficiently brilliant (or devious) student could fill the entire archive with "guess-type" solutions of one step if such solutions were not automatically excluded! Similarly, a problem terminated in the give-up trapping state must not be considered as improving the archive.

## The Information Updating and Student Learning Models

As was shown in Chapter 3, between learning discontinuities the student statistics may be updated in a very simple way. The Dirichlet distribution allows us to simply add the number of responses in each category to the corresponding exponent in the form of the distribution. One needs only store a matrix M of these exponents to completely characterize the distribution. Specifically, if technique $T_j$ is chosen for a problem in state $s_i$, then matrix element $m_{ij}$ undergoes the

updating

$$m_{ij} \text{ <-- } m_{ij} + 1$$

The use of the student learning model, on the other hand, is more challenging. In phases 1 and 2 of the research, we had assumed that the student's learning took place more or less continuously; and did not anticipate recognizing any sudden shifts in the student's problem solving patterns measurable over the span of a single problem. Because of these assumptions, a very simple model was adopted for updating the student's patterns. It was assumed that the last N responses in each state would be the most relevant representation of the student's patterns. We had hoped to find an estimate for the optimal value of N that would balance the loss of statistical "weight" from a small sampling with the increase of relevancy of looking only at the most recent responses. Such an optimal value would depend presumably upon some sort of "learning rate" characteristic of the process.

Unfortunately, the second round of student measurements revealed unmistakeable indications that the students changed their problem solving patterns suddenly and at unpredictable intervals. This evidence will be presented and discussed in the next chapter, but this important result is mentioned here to explain why the final student learning model differs so much from the model first used with the students. We see, in particular, that a student "history" of N responses cannot model a sporadically changing set

of response probabilities realistically. On the other hand, thanks to the observed correlation of the learning discontinuities with occupancy of student failure states, we can now identify the moments at which to apply the student learning model developed in Chapter 2. Although the tutor was undoubtably making some sub-optimal problem choices (based on the student history model), the complete record of each student's responses over a wide variety of problem classes is still available, and thus allows us to measure the parameters of the new learning model from the raw data.

## Description of the Computer Tutorial System

The tutorial system is written in LISP 1.6, a dialect of LISP developed at the Stanford Artificial Intelligence Laboratory by John McCarthy and colleagues. Since a typical tutorial session involves substantial algebraic manipulation, the tutor depends upon the resources of a comprehensive algebraic package called REDUCE written in LISP by A. C. Hearn of the University of Utah. The tutor calls the command scanner in REDUCE to read every formula and return the LISP prefix equivalent. Although some minor formatting cleanup is done by the tutor, all algebraic manipulations including differentiation are sent to REDUCE. REDUCE sends back the resulting simplified expression in LISP prefix notation, modified in form by various flags that are selected by the tutor. Finally, when expressions are printed out

to the student, the tutor calls a REDUCE program to format the individual expression terms. All other manipulations, including processing of student word responses, manipulation of the student models, variable substitution, trigonometric substitution, integrating by parts, polynomial division, trigonometric identities, and partial fraction expansion are done by the tutor.

# Chapter 5

## Experimental Results

This chapter describes two experimental episodes with students learning calculus and sketches the limitations of each experiment and the lessons learned. A detailed justification of the assumption of the existence of learning discontinuities is derived from examination of the students' responses. Numerical results from the second episode showing the student's expected number of steps as a function of the tutor's expected length of solution are then presented. We shall examine the student's probability of entering a failure state as a function of the number of problems worked and shall estimate the mean of the student's learning parameter $\alpha$. Finally, the results of the tutor optimization are presented.

### The First Experiment

A group of four college freshman calculus students was chosen to help debug the prototype tutor. Although the students were serious in their desire to learn techniques of integration from the tutor, the experiment itself was a qualitative test of the integration routines and the mode of interaction with the students. Other than the identification of the computer program bugs, the principal impressions gained from the students were:

1. The need for an archive of problems from which the

tutor can select examples. The students were typically reluctant to suggest more than a few of their own examples and of course lacked the perpsective to choose those examples most beneficial to their development.

2. The need for the technique of "guessing the answer" so that the student could circumvent simple but repetitive patterns. The students also enjoyed the challenge of guessing occasionally when they understood which technique to apply, particularly with simple substitution and simple trigonometric substitution.

## The Second Experiment

Following the preliminary experiment, the general problem archive and the guessing technique were added along with a number of minor alterations to the tutor's conversational format. Facilities for recording each student's response were added so that complete protocols could be reconstructed. Fifteen students from Stanford University voluntered to interact with the tutor over a period of about three weeks. No particular attempt was made to screen the students for a certain type of background, although all the students were either studying calculus concurrently or had studied calculus in the past and were interested in resurrecting their skills at methods of integration. In short, the students exhibited the reasonably broad spectrum of prior mathematical expertise that a tutor would expect to

encounter.

During the course of the experiment the students worked a total of 284 problems (19 per student) of which 258 (91%) were selected by command of the students from the tutor's problem archive. 282 of the problems (99%) were terminated in the solved state, and the other two were unsolvable problems initiated by the students (e.g. $\int e^{X^2} dX$). The guessing technique was used 45 times with a success rate of 89%. The students asked the tutor for direct help in 90 of the problems. Typically once help was requested, it was requested repeatedly. In the 90 "helped" problems, the students asked for technique choosing assistance 173 times and technique application assistance 65 times. The students entered identifiable failure states (where application of the trial technique failed to yield a new transformation) 98 times on 65 different problems. The probability that the student would enter a failure state was 0.29 if he had not previously entered a failure state on that problem and 0.40 if he had already entered a failure state on that problem. The probability that the student would ask for help was 0.32 if he had not entered a failure state and 0.54 if he had entered a failure state on a particular problem.

## Determination of the Failure Parameter

In Chapter 2 we defined the failure parameter $\theta$ as the amount by which the probability of choosing a technique decreased

given that the student had chosen the technique on the previous step and encountered a failure. An accurate estimate of $\theta$ was difficult to make since the failing technique was rechosen only 7 times out of the 98 failures encountered. Furthermore, 5 of the 7 reapplications of failing techniques involved substitution, while the other two were trigonometric identities.

Averaging over all the students, we found the results expressed in Table 5.1:

| State | Technique | $t_{ij}$(nonfailure) | $t_{ij}$(failure) | $\theta(i,j)$ |
|---|---|---|---|---|
| 4 | subst. | 0.275 | 0.167 | 0.61 |
| 7 | subst. | 0.755 | 0.500 | 0.66 |
| 13 | subst. | 0.291 | 0.222 | 0.76 |
| 4 | trig. iden. | 0.217 | 0.667 | 3.06 |
| all others | | $t_{ij}$ | 0 | 0 |

Since the substitution failures were made by several students, a value of $\theta = 0.7$ seems reasonable for this technique. The large $\theta$ value for trigonometric identities is questionable since it is based on only 2 responses made by the same student.

## The Existence of Learning Discontinuities

A major assumption in Chapter 2 was that learning occured suddenly and at unpredictable intervals. This assumption allowed us to separate neatly the information updating and student learning processes. We assumed furthermore that we could identify the occurrences of student learning unambiguously, thus knowing when to apply the student learning model. We shall now present an

analysis of the students' responses that makes our assumptions of the existence and properties of learning discontinuities more credible.

Consider an experiment in which we the observers have only the power to observe the responses made by the participants. We are to assume nothing about the purpose of the experiment or the meaning of the responses. The responses themselves are sequences of positive integers which we assume arise from a multinomial distribution. We are told by the designers of the experiment that at certain designated points in the sequences it is likely that the participants altered their rationale for responding. The suspicion of uniqueness of these points arises from observations that we are not permitted to see. We are asked to analyze the response data to a) support or reject the hypothesis that the suspicious points separate differing reponse regimes; and b) test the inclusiveness of the experimenter's criterion for selecting suspicious points by trying to find additional points that are significant statistically as regime separators. In this hypothetical experiment we have purposely obscured the underlying rationale for "suspecting" a given point so as not to allow the observer any bias in deciding that such a point indeed ought to separate response regimes.

In order to answer question a, we propose to consider the sequence of responses $s_1$ before each suspicious point and the sequence of responses $s_2$ following each point. Using these

sequences we shall calculate the chi-square statistic for the particular suspicious point. The chi-square statistic is chosen since it is the natural comparison statistic for independent samples from two multinomial distributions. Since the magnitude of the chi-square statistic depends on the sample size, each candidate pair of sequences will be compared to 1000 sequences randomly generated with the same overall response probabilities. We shall take as the null hypothesis the event that the subsequences $s_1$ and $s_2$ do not arise from different multinomial distributions. Thus if sequence $s_2$ really does represent a statistically significant change from sequence $s_1$, the resulting chi-square statistic will be large in comparison with most of the 1000 sequences generated under the null hypothesis. In practice, we shall accept only those suspicious points whose chi-square statistic has a significance of 90% or more (whose chi-square statistic is strictly greater than 90% of the chi-square values generated by the null hypothesis). Once we have identified a point successfully as separating two regimes of responses, we must ignore sequence $s_1$ in examining points further along the data since responses from sequence $s_1$ will contribute falsely to raising the chi-square values of subsequent points.

To answer question b, we shall repeat the calculation of the chi-square statistic and the 1000 null hypothesis trials at all of the nonsuspicious points to see how many "non-suspicious" points are also regime separators. This is crucial as a test of

the model predicting the occurrence of discontinuities.

Of course, this hypothetical experiment describes exactly the situation we face when trying to identify the learning points from the calculus students' response data. The suspicious points are those places where the student encountered a failure state and presumably had to consider whether or not his solution schemes were practical. As emphasized above, we did not make any assumptions about the student data other than assuming that between learning events each student's reponses were derived from a multinomial distribution. This was felt to be a fair test of the existence of learning points since inclusion of extraneous student entries and ad hoc interpretation of each student protocol was avoided.

Several interesting facts were uncovered by this search. As a general rule, a minimum of eight responses are needed to establish a 90% certainty of the existence of a learning point, even with the most extreme data. For instance, the sequence 1 1 1 1 2 2 2 (consisting of only seven responses) does not possess any division into subsequences, even after the fourth response, that generates a chi-square statistic with 90% significance. In a similar vein, regardless of the total length of the sequence, the first two responses are incapable of indicating a learning point. For instance the sequence

1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

also does not possess any division into subsequences that generates a chi-square statistic with 90% significance. This result has the incidental effect of causing most of the changes in response probabilities due to "the start-up transient" not to be considered as significant learning points. (Nearly all the students made one or two anomalous responses at the outset before they became familiar with the tutor).

The student responses were separated by state and the suspicious points were identified by looking at the complete protocols and marking all the times a student applied a transformation that failed to yield a new integrand (definition of the failure state). After response lists of fewer than eight responses and failure states occurring in the first two responses were eliminated, a total of 37 suspicious points remained. The chi-square analysis showed that 17 of the 37 points (45.9%) were indeed significant as response regime separators at the 90% level. Six more were significant at the 80% level, but this is only mentioned to show that most of the insignificant points were very insignificant! The most important result of this analysis was that a complete scan of all the responses (461 in all) produced only three additional points significant as response regime separators at the 90% level. Thus although only 45.9% of the suspicious points seem to be genuine learning points, 85.0% of all possible learning points are identified by our model.

Why are half of the student failure states obviously not

learning points? A detailed examination of the student protocols for each of the insignificant points shows that the contributing causes are diverse. In three cases the failure state was "false" since the student subsequently reapplied the same technique successfully. In at least 10 cases the student encountered a succession of failure states in more than one problem type. Since the tutor tended to choose archive problems from the most "critical" problem classification, some students did not return to all the troubled states frequently enough to produce a reasonably long run of failure free responses. Several short sequences of responses were encountered that were interspersed with two or more failure states and yielded inconclusive results. This, of course, should be viewed as a mild failure of the experiment since in this case it is not clear whether the student finished the experiment too soon or whether the tutor failed to teach the student effectively.

Returning to the original question of this section, what was the distribution of identified learning points in the student's responses? Examining 44 subsequences generated from the students' set of responses divided at each learning point (some student response sequences possessed no learning points), we find that the average number of responses generated between learning points is 461/44 = 10.47, but the the distribution of this number varies from 3 to 39 with 18 different values measured. Referring to Figure 5.1, we can now draw the conclusion that the response

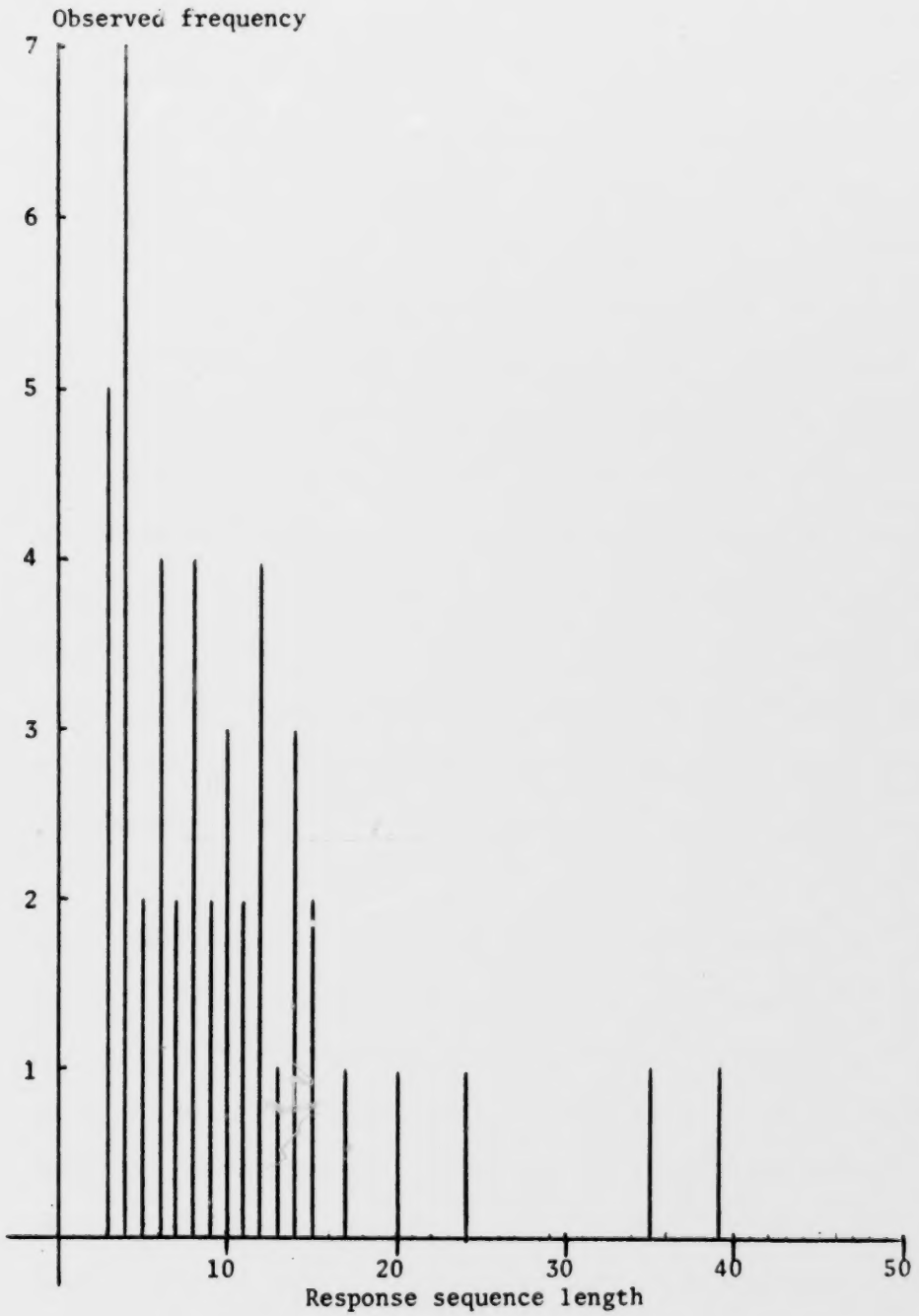


Figure 5.1. The number of reponses between learning points.

discontinuities are sudden since our analysis shows that 85% of all the significant learning points agree with our model of the failure state as being the precise point where the responses change significantly. Furthermore we are justified in claiming that the discontinuities occur sporadically since we have just seen that the average number of responses between learning points is widely distributed. We have thus given a strong argument for the existence of response discontinuities and in this context have justified the separation of the information updating and student learning models that we performed in Chapters 2 and 3.

## The Student's Expected Number of Steps

The most interesting question to ask about the calculus tutor is whether the students became better problem solvers after exposure to the tutor. As we have explained, there are many possible criteria of problem solving excellence. For instance, elegance might be defined in terms of the use of certain very general problem solving techniques. This project has focused on solution length as a reasonable criterion. If the students "solve problems in a fewer number of steps after exposure to the tutor" then the students have profited in a measurable way. Unfortunately, the number of steps to solution is not a fixed property of a given problem state (except for the special states #1, #2, and #3). We did use the tutor's expected number of steps to solution (plus a factor depending upon the variance of the

expected number of steps) in defining the problem length threshold for each state, but we recognized that this was only a guide to help the tutor identify most of the unwieldy solutions. If we insist on a very accurate measurement of the student's expected number of steps, we must realize that the measurement depends largely on the particular problems the student chose to work. However since nearly all of the students' problems were selected from the problem archive, we can compare the length of the students' archive solutions to those of the tutor as a function of number of problems worked to define a measure of improvement for the student. Again it may be argued that whether or not the student can approximate closely the tutor's length of solution depends upon the particular problem, but we shall assume that these effects are not significant.

Figure 5.2 shows a plot of the students' average number of additional steps per problem versus the tutor as a function of the number of problems worked. For each number of problems worked the single highest instance of additional steps was ignored. This was done because the raw data included several anomalously long solutions all of which turned out from direct examination of the problem protocols to be instances of the students experimenting with features of the computer tutor! Notice the general downward trend of the points, indicating that the students gradually learned how to solve problems in as few steps as the tutor. No negative entries are recorded here since each time a student

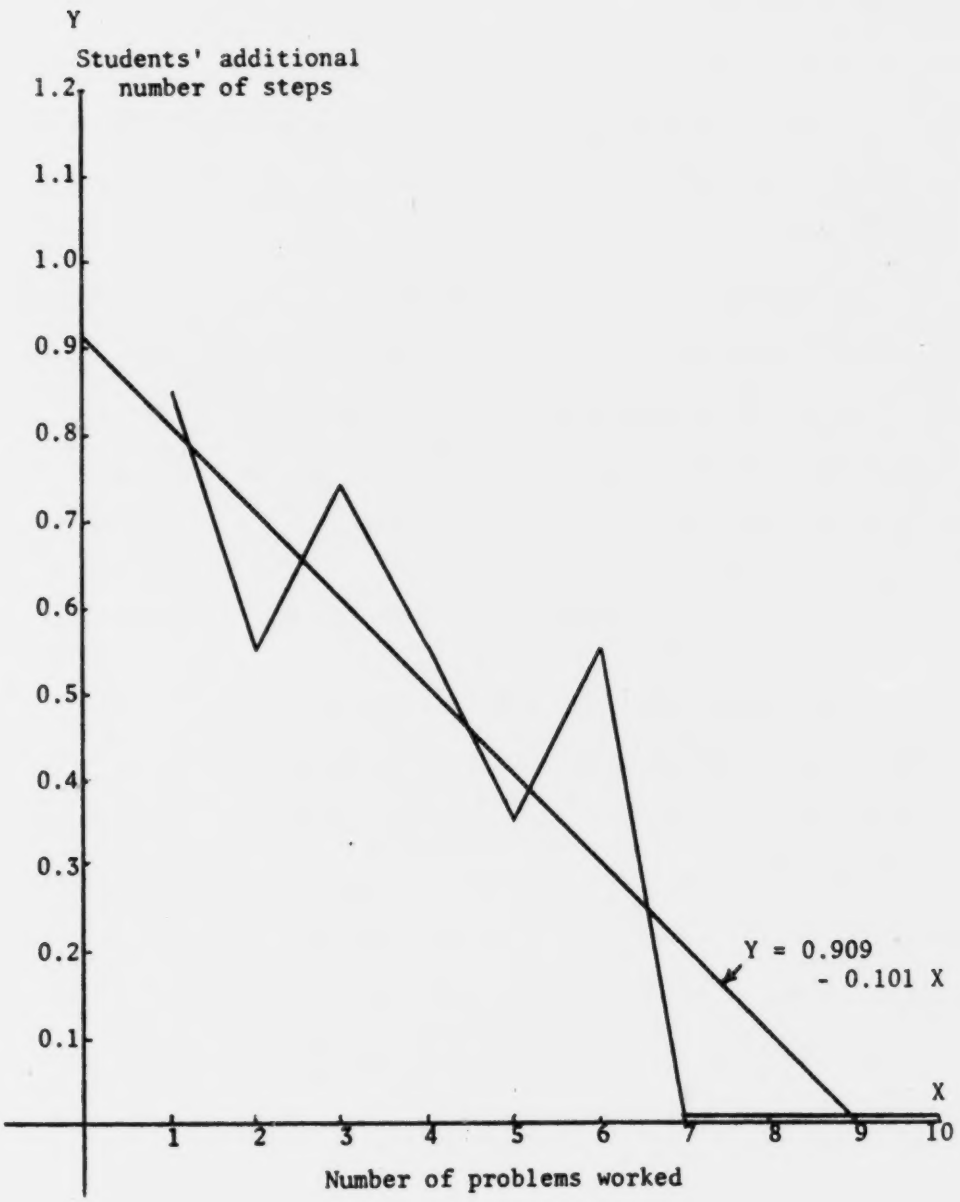


Figure 5.2. The student's average number of additional steps per problem worked as a function of the number of problems worked.

produced a shorter solution than the tutor, the tutor incorporated the solution into its own problem solving patterns. The results thus show the performance of the students compared with the fully optimized tutor that existed at the end of the experiment. Although we know that the data should not be expected to be linear with the number of problems worked since we have just discussed the abrupt nature of the typical learning pattern, we shall take the liberty of representing the data averaged over all the students by a least squares linear fit in order to point out its basic properties. This linear fit is given by

$$Y = 0.909 - 0.101*X \quad .$$

The interesting part of this equation is the slope of -0.101, which indicates that the student comes 0.101 steps closer to the tutor each time the student works a problem. Notice that this indicates that the students, on the average, become as proficient as the tutor after working about nine problems in each problem class. An exponential model would be a better fit, but the above result gives an indication of how a "learning per problem" quantity can be measured.

## The Probability of Entering a Failure State

Another quantity related to gaining problem solving expertise is the probability of entering a failure state. We have already showed that the students converge on the tutor at the rate

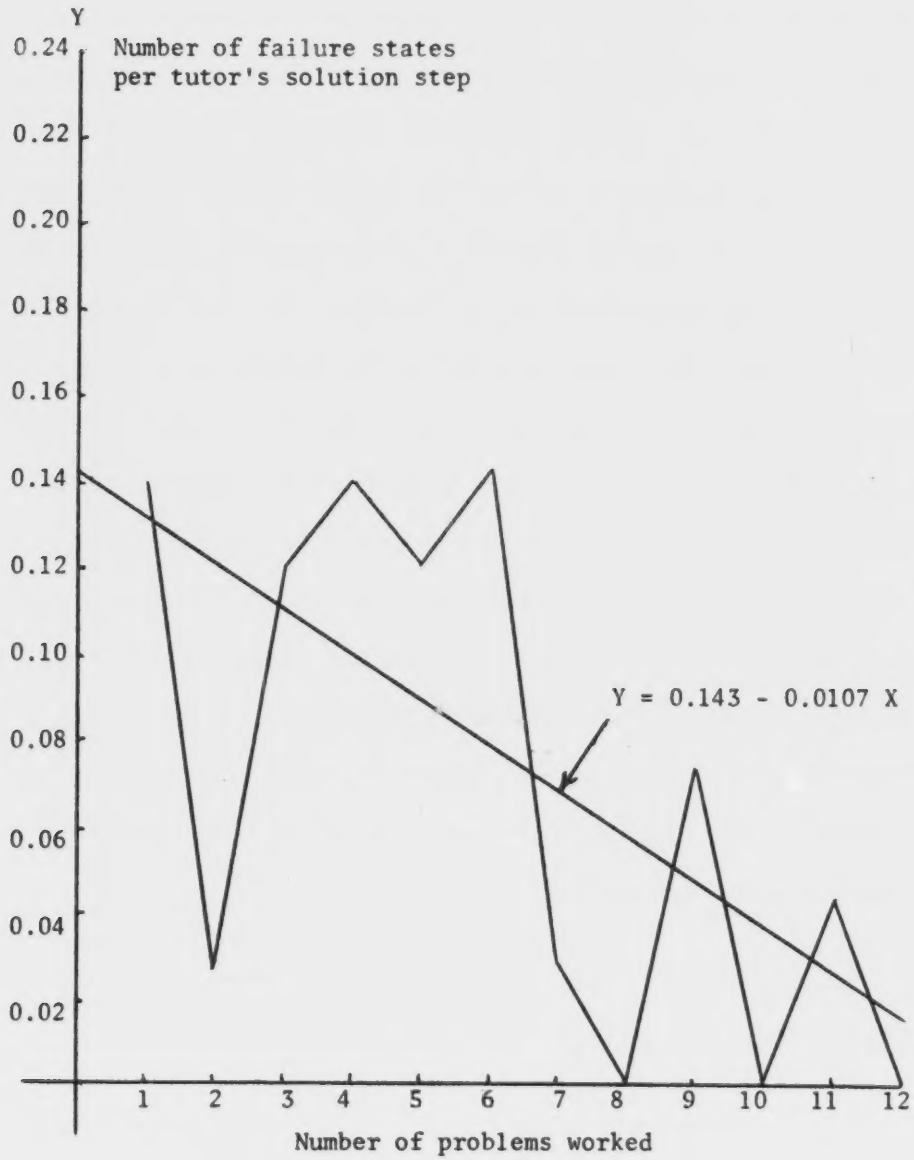


Figure 5.3. The number of student failure states per problem per tutor step as a function of the number of problems worked.

of about 0.1 steps per problem worked. If in addition, the students reduce their probability of entering a failure state as a function of the number of problems worked, we can be reasonably sure that they are really learning to solve problems more efficiently. In Figure 5.3 we show the number of failure states encountered on a problem divided by the number of steps used by the tutor to work the problem as a function of the number of problems worked. We have divided by the length of the tutor's solution in order to correctly scale the difficulty of the problems (remember from Chapter 4 that the tutor generally chose the shorter, and thus easier, problems from the archive first). The data is too noisy to draw many conclusions, but a clear trend downward is seen after about six problems worked. The peak between 4 and 6 problems worked is likely due to the increased difficulty of a few particular problems usually encountered at that point. For instance, after working one or two simple substitution problems like

$$\int \sin(2*X)\ dX \quad \text{and} \quad \int \cosh(X/4)\ dX\ ,$$

nearly all of the students got

$$\int \cot(2*X)\ dX$$

as the next problem. Although the tutor knew that the command TRIGIDEN would change this to the more suggestive form

$$\int \frac{\cos(2*X)}{\sin(2*X)}\ dX \quad ,$$

most of the students became confused after finding out that

$$\int \cot(Y)\ dY$$

was not a "known" integral and tried unusual substitutions or trigonometric identities before realizing how simple the problem was. The result of all this is a peak in most of the measured statistics wherever this problem appeared. This problem remains as a good example of how hard it is to assign a consistent difficulty factor to integration problems.

For descriptive purposes a least squares linear fit to the data in figure 5.3 yields the relation

$$Y = 0.143 - 0.0107\ X \quad .$$

This indicates that the number of student failures per step decreases by 0.0107 for each successive problem the student works.

## Estimation of the Learning Parameters

Chapter 2 proposed a scheme for altering our estimates of the student's technique response probabilities when the student encountered a learning discontinuity. Since we have shown in this chapter that we can identify (with probability 1/2) those moments when the student actually does encounter the discontinuities, all that remains is to deduce realistic numerical parameters for the

model from the student data.

The basic assumption made in Chapter 2 was that when a learning discontinuity occurred, only the techniques chosen immediately before and immediately after the discontinuity had their response probabilities affected. The model assumed that there was a learning parameter $\alpha$ such that if the student encountered a learning discontinuity in state $s_i$ as a result of applying technique $T_j$ and then subsequently applied technique $T_k$ sucessfully, then

$$t_{ij,posterior} = \alpha t_{ij,prior}$$

and $$t_{ik,posterior} = \eta t_{ik,prior}$$

where $\alpha$ is beta distributed with parameters r and s. $\eta$ depends on $\alpha$ and is given in Chapter 2. Examining each of the 20 confirmed points of learning discontinuity, we calculate the estimates

$$\bar{\alpha} = \frac{1}{20} \frac{t_{ij,posterior}}{t_{ij,prior}} = 0.169$$

and similarly,

$$v_\alpha = \frac{1}{20} \left(\frac{t_{ij,posterior}}{t_{ij,prior}} - \bar{\alpha}\right)^2 = 0.086$$

where $t_{ij,posterior}$ is estimated by the observed response frequencies.

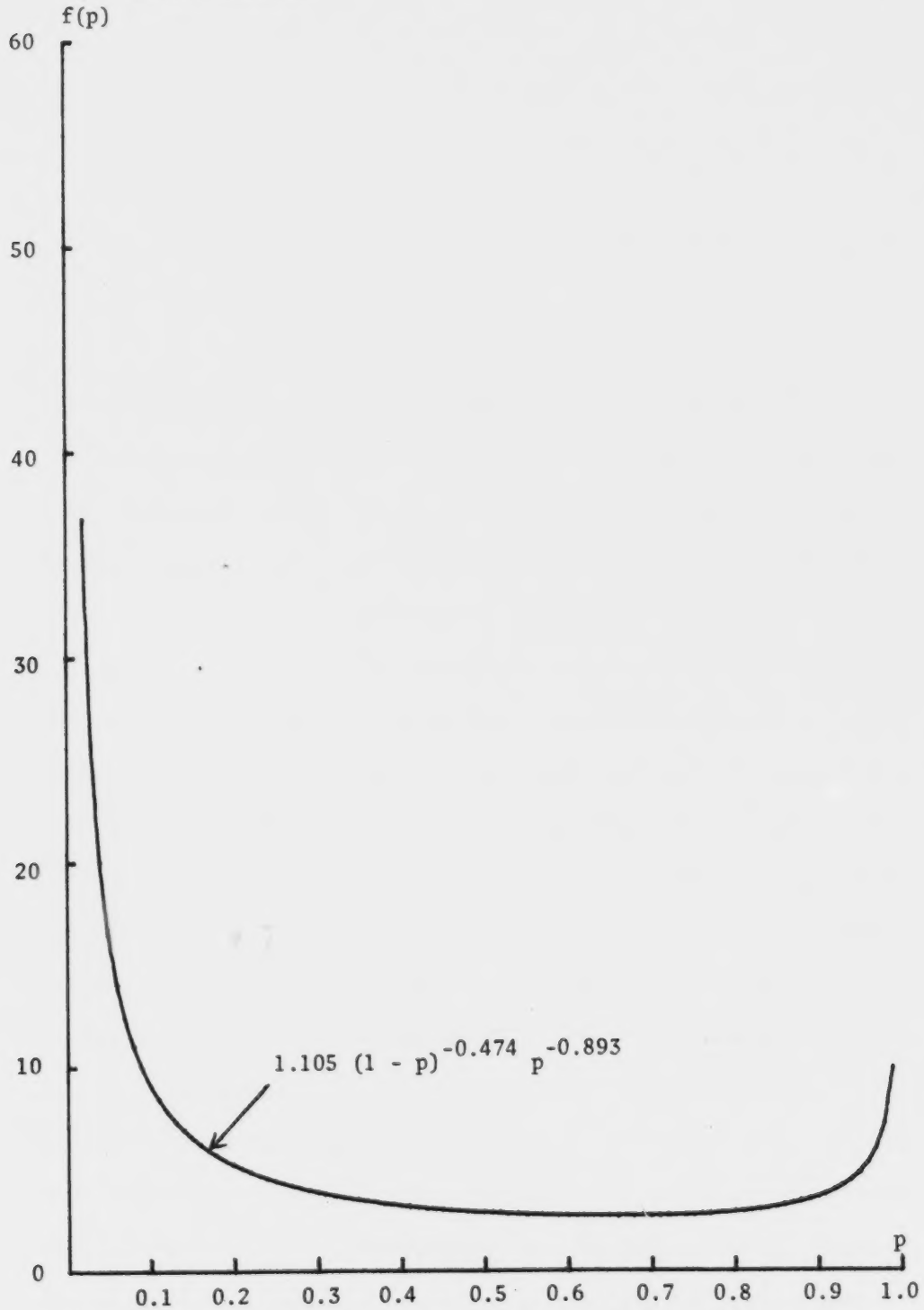


Figure 5.4. Distribution of $\alpha$ for the calculus students.

Since

$$\overline{\alpha} = \frac{r}{r+s} \quad \text{and} \quad V_{\alpha} = \frac{r * s}{(r+s)^2 * (r+s+1)}$$

we can use the methods of moments and solve for r and s to get

$$r = 0.1070 \ , \ s = 0.5260$$

The graph of this beta function is shown in figure 5.4. Now that we have the actual form of the distribution for a reasonably large group of students, we shall consider these parameters as describing uncertainty in the student learning parameters for all such events in the future.

A relevant question at this point is whether many of the other student technique probabilities changed besides the techniques $T_j$ and $T_k$ specifically mentioned in the model. Calculating "$\alpha$" parameters for all the remaining possible responses in each case, we find that the technique applied immediately before the failure had the smallest $\alpha$ and the technique applied immediately after the failure had the largest $\alpha$, as the model predicts. The observed $\alpha$'s for each position from 5 reponses before the failure to 5 responses after the failure are shown in figure 5.5. Each dot represents a single measurement of $\alpha$ in a particular postion before or after the failure event. Notice the remarkable discontinuity in the observed $\alpha$'s before and after the failure. An unexpected observation is that the techniques applied two and three positions

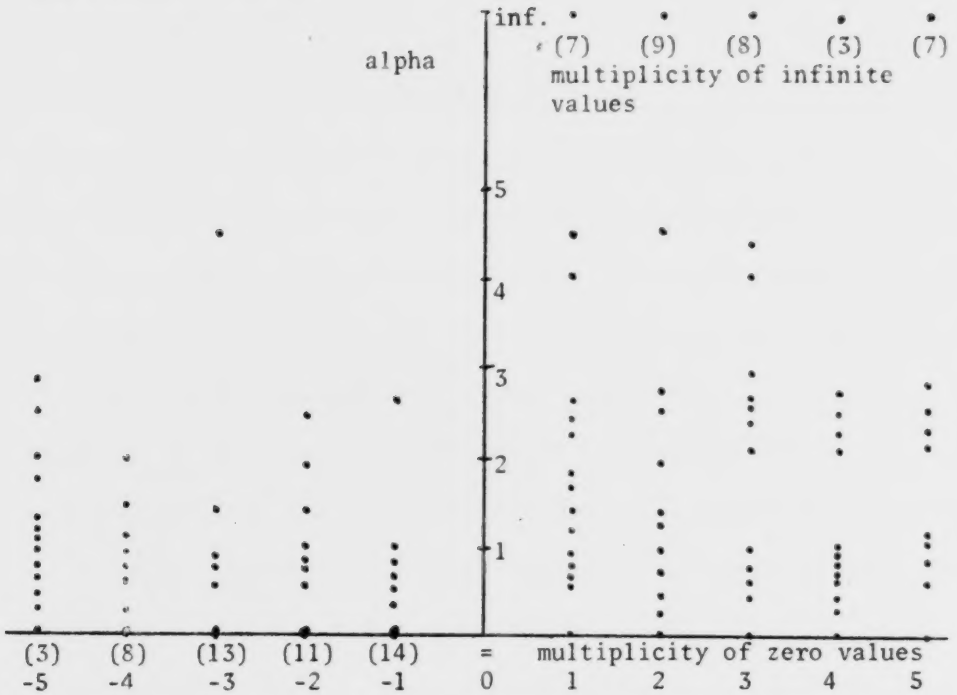


Figure 5.5. Observed $\alpha$'s, dependent positions included.

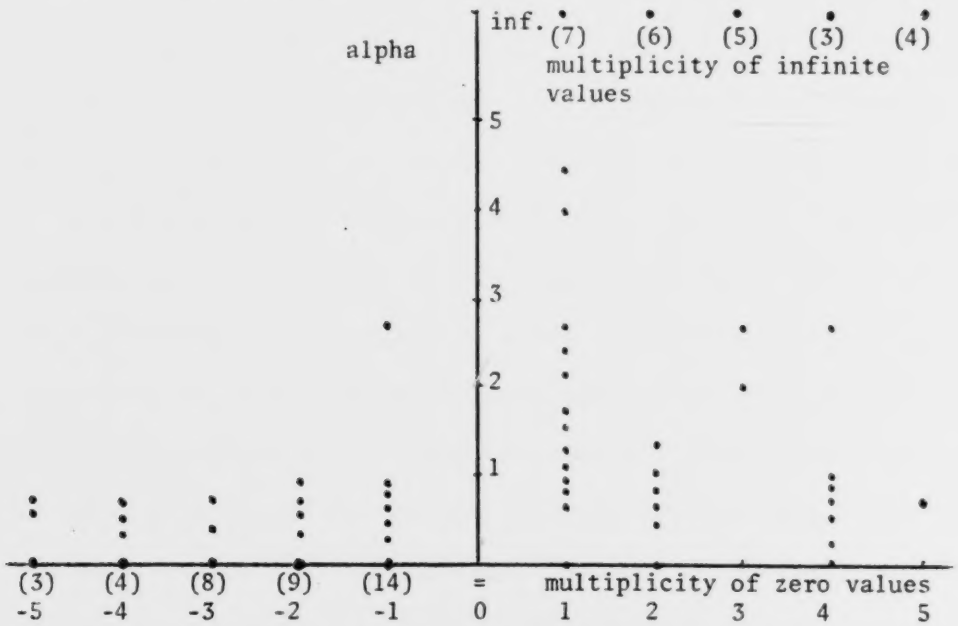


Figure 5.6. Observed $\alpha$'s, dependent positions removed.

away from the failure also seem to be affected. This would seem to seriously undermine the assumption that only the techniques adjacent to the failure are affected. However, it is not true that in this data the positional observations are independent. For instance, in many cases the pre- or post-failure technique is also applied in other positions. Subtracting these occurrences reduces the correlation effect but does not cause it to disappear as shown in Figure 5.6. The conclusion is that the failure state does seem to affect the other techniques applied "nearby" in addition to the ones predicted by the model. A systematic inclusion of these other techniques seems difficult since there is no obvious rationale for their technique frequencies altering as a result of the failure. Possibly after discovering a new technique, the student is stimulated to think about his problem solving patterns or is more prone to experiment with new techniques. In any case, the measurements indicate a definite tendency for a few techniques before the failure to decrease dramatically in frequency after the failure, and conversely a few techniques after the failure increase dramatically in frequency. As the figures show, the largest effect is the technique predicted by our student learning model. We view this result as a qualified success with interesting implications for future research. Further work with this problem would be aided by longer student sequences and possibly direct interviews with the students to establish a rationale for prediction.

Definition of a Learning Rate for Tutorial Systems

The above has illustrated some of the difficulties in defining a learning rate for the methods of integration system. Of course, this difficulty stems from the tutorial nature of the system rather than from the subject of integration. In fact, the subject of integration probably allows measuring a learning rate more easily than other subjects since the notion of problem transformation is so simple to define.

The central idea of this tutorial system is that within each problem classification, the tutor's problem solving strategies are determined on a frequency basis. As we have mentioned, because of this approach, the tutor never has to find a solution and thus never knows how hard an individual problem is. Great advantages in the actual tutoring process accrue from this approach; for instance, the tutor can deal with problems it has never worked, it can learn from the students, and it can adjust its problem solving techniques to the level of the learner. But these advantages have a price since the tutor has no absolute standards against which to judge the student. If an advanced student logs in to suggest only difficult problems to the tutor, the resulting statistics may be the same as a beginning student who has tried to work easy problems! Only the fact that the overwhelming majority of the problems chosen by the students in the experiment described in this chapter were from the tutor's problem archive made the analysis of the learning rate meaningful.

## The Tutor Optimization Experiment

Before the methods of integration experiment the author believed that one or two of the students might be so adept that they would actually construct shorter solutions to some of the 73 archive problems. Since the author had been involved in integral problem solving for at least two years prior to the experiment, and considered himself an expert integral solver, there seemed little chance that any improvements would actually occur. It was his intention to implant one or two "doctored" solutions in the archive to see if they were improved upon. However, this plan was overlooked in the exigencies of getting the tutor running and the students organized. Upon examining the archive at the end of the experiment, it was found that no less than 18 of the original problem solutions had been shortened! From the detailed solution schemes, it was apparant that the tutor had acquired technique patterns never before used by the author. This was a lesson of the first magnitude.

The scope of the improvement was also unexpected. Of the 11 problem types represented in the archive, four were improved significantly. Table 5.2 shows the average number of problem steps for the tutor before and after the experiment, broken down by problem type. See Chapter 4 for the description of problem types.

| Type | Description | Before | After | Change |
|---|---|---|---|---|
| 1 | known | 1.000 | 1.000 | 0 |
| 2 | simple substitutions | 2.000 | 2.000 | 0 |
| 3 | simp. trig. subst's | 2.447 | 2.000 | -0.447 |
| 4 | trig. functions | 2.857 | 3.000 | +0.143 |
| 5 | exponential fns | 2.250 | 2.250 | 0 |
| 6 | arctrig. functions | 4.250 | 4.368 | +0.118 |
| 7 | fract. poly. powers | 4.470 | 3.320 | -1.150 |
| 8 | comb. of 4 and 5 | 1.000 | 1.000 | 0 |
| 9 | comb. of 4 and 7 | 3.000 | 2.000 | -1.000 |
| 12 | comb. of 6 and 7 | 5.250 | 5.368 | +0.118 |
| 13 | quotients of poly's | 6.805 | 5.421 | -1.384 |

The largest drop was for quotients of polynomials, type 13. Notice that three categories experienced slight gains, indicating that in the new problem solving scheme a trade-off between categories occurred. It is also interesting to examine the changes in the $t_{ij}$'s to see what techniques the improved tutor is more likely to use. Table 5.3 gives the values of

$$t_{ij,after} - t_{ij,before}$$

for relevant values of i and j.

| type | technique | change in application frequency |
|---|---|---|
| trig. fns. | deriv. subst. | +0.037 |
| " | parts | +0.009 |
| " | trig. subst. | +0.018 |
| " | trig. ident. | -0.064 |
| frac. poly. powers | deriv. subst. | +0.111 |
| | trig. subst. | -0.111 |
| quotients of polynomials | deriv. subst. | +0.100 |
| | trig. subst. | +0.002 |
| | sum separation | -0.093 |
| " | poly. division | -0.004 |
| " | compl. square | -0.020 |
| " | part. frac. exp. | +0.015 |

For problems of type 4 (trigonometric integrands) the tutor now uses trigonometric identities less in favor of derivative substitution, integration by parts, and trigonometric substitution. For fractional powers of polynomials the tutor now recommends derivative substitution more often in place of trigonometric substitution. Finally, for quotients of polynomials the tutor now essentially recommends derivative substitution in place of separation of the sum. This last change is one that had never occurred to the author. For instance, with the integral

$$\int \frac{X^2 + 2*X + 5}{X - 4}\, dX$$

rather than dividing out the polynomials or separating the sum into three integrals, it is shorter to substitute $U = X - 4$, yielding

$$\int \frac{U^2 + 10*U + 29}{U}\, dU$$

which is solved immediately by inspection as

$$\frac{U^2}{2} + 10*U + 29*\log(U)$$

which upon resubstitution is

$$\frac{X^2}{2} + 6*X - 32 + 29*\log(X - 4)$$

The unpredictable occurrence of better solutions is an interesting feature. Seven different students contributed to

optimizing the archive, including students who otherwise appeared to be the least proficient of the integral solvers.

The potential of the tutor's self improving scheme is great. One may wish to carry out the optimization simultaneously over several superiority criteria and several levels of student sophistication. But perhaps most important, the tutor did improve those areas in which the tutor's original author was weak.

## Conclusions

This research has extended and deepened the definition of a tutor in computer-based education. In particular, the tutor transmits problem solving heuristics, chooses appropriate examples, deals with arbitrary student examples, handles diverse student backgrounds, and learns superior problem solving heuristics from the students.

A logical and quantitative methodology for transmitting problem solving heuristics has been established. The use of problem archives as the basis of the tutor's own heuristic schemes is demonstrated.

A simple model is posed of how student heuristics change when the student encounters a failure and is supported by experiment with calculus students.

A definition of learning in a tutorial situation is given and is demonstrated by the calculus students.

Perhaps the most interesting result of the research is the

scheme for tutor improvement. In the calculus experiment the tutor not only acquired a larger number of improved problem solutions than had been expected, but incorporated problem solving strategies previously unknown to the author.

Finally, this research combined for the first time the results of recent research in symbolic integration (Moses, 1967) and algebraic simplification (Hearn, 1970) for use in computer assisted instruction.

## Recommendations for Future Research

This research has suggested a number of interesting new directions for future work. At the heart of the student learning model, much more needs to known about the role of a failure in determining the student's technique choice probabilities. An unexpected result of the calculus experiment was that techniques other than the failing one are apparantly affected by the failure.

Experimentally it was shown that calculus students' solution lengths converged to the tutor's solution lengths in about 9 problems. How much of this convergence is attributable to learning how to interact with the computer tutor and how much represents true changes in the student's problem solving strategies?

Taking a different approach, a utility theory of problem presentation could be implemented that used the expected rate of student convergence in different problem classes as a criterion.

The use of a quantitative measure of problem difficulty was avoided completely in this research. The development of a good "difficulty metric" for integration problems that did not involve searching for a solution explicitly would, of course, be a significant result in artificial intelligence as well as computer assisted calculus tutoring.

Finally, the most obvious new direction for computer assisted tutoring is developing tutors for subjects other than integration. Since methods of integration can be modeled by a simple state and technique structure, construction of tutors for other subjects will undoubtably deepen the understanding of states and techniques. Methods of integration also involves a very traditional problem solving structure with heuristics being a dominant component. These heuristics are clearly revealed in the automatic integration programs and in the integration tutor. How dominant the role of heuristics is in other subjects is not known.

## Bibliography

Alpert, D., & Bitzer, D. L. Advances in computer assisted instruction, Science, Mar 20, 1970, 167, 1582-1590.

Atkinson, R.C., Bower, G.H., & Crothers, E.J. An introduction to Mathematical Learning Theory, New York: Wiley, 1965.

Bobrow, D. G. Natural language input for a computer problem-solving system, in Semantic Information Processing, (Ed.) Minsky, Boston: MIT press, 1968, 146-226.

Carbonell, J. R. AI in CAI: An artificial-intelligence approach to computer assisted instruction, IEEE Transactions on Man-Machine Systems, 1970, MMS-11, 190-201.

Carbonell, J. R. Mixed-initiative man-computer instructional dialogues, BBN report no. 1971, Cambridge, Massachusetts: Bolt, Beranek and Newman, Inc., May 1970.

Colby, K., Tesler, L., & Enea H. A directed graph representation for computer simulation of belief systems, Computer Science department Report, 1967, Stanford University.

Colby, K., & Enea, H. Inductive inference by intelligent machines, Computer Science department Report, 1967, Stanford University.

Colby, K., & Enea, H. Heuristic methods for computer understanding of natural language in context restricted dialogues, Computer Science department Report, 1967, Stanford University.

Colby, K. & Smith, D.C. Dialogues between humans and an artificial belief system, Stanford Artificial Intelligence Project Memo AI-97, 1969, Stanford University.

de Groot, M. Optimal Statistical Decisions, New York: McGraw-Hill, 1970.

Engleman, C. Mathlab: A program for on-line machine assistance in symbolic computations, Report MTP-18, The MITRE Corporation, Box 208, Bedford, Massachusetts.

Freeman, F. R. Medical diagnosis: comparison of human and computer logic, Bio-Medical Computing, 1972, 3, 217-221.

Feurzeig, W., Wexelblat, P., & Rosenberg, R. C. SIMON - A simple instructional monitor, IEEE Transactions on Man-Machine Systems, 1970, MMS-11, 174-180.

Grandey, R. C. An investigation of the use of computer-aided instruction in teaching students how to solve selected multistep general chemistry problems. Unpublished doctoral dissertation, University of Illinois, 1970.

Hearn, Anthony C. Reduce 2 user's manual, Stanford Artificial Intelligence Project Memo AIM-133, 1970, Stanford University.

Hearn, Anthony C. The problem of substitution, Stanford Artificial Intelligence Project Memo AI-70, 1970, Stanford University.

Howard, R. A. Dynamic Probabilistic Systems, New York: Wiley, 1971.

Leonard, G. B. Education and Ecstasy, New York: Delacorte Press, 1968.

McCarthy, J., et. al. LISP 1.5 Programmer's Manual, 1965, Computation Center and Research Lab of Electronics, Cambridge: MIT Press.

Moses, Joel Symbolic Integration, doctoral dissertation, Cambridge, Massachusetts: Project MAC report, 1967.

Ostrowski, A. Differential and Integral Calculus, 2nd ed., Chicago: Scott, Foresman and Co., 1968.

Papert, S. Teaching children thinking, Programmed Learning and Educational Technology, Sept 1970, 9, 5, 245-255.

Papert, S. and Solomon C. Twenty things to do with a computer, Educational Technology, April 1972, 12, 9-18.

Postman, N. & Weingartner, C. Teaching as a Subversive Activity, New York: Delacorte Press, 1969.

Slagle, J. R. A heuristic program that solves symbolic integration problems in fresman calculus, in Computers and Thought, (Eds.) Feigenbaum & Feldman, New York: McGraw-Hill, 1963, 191-206.

Smallwood, Richard D. a Decision Structure for Teaching Machines, Cambridge, Massachusetts: MIT Press, 1962.

Thomas, George B. Calculus and Analytic Geometry, 3rd ed., Massachusetts: Addison-Wesley, 1961.

Weizenbaum, J. ELIZA: A computer program for the study of natural language communication between man and machine, Report, 1965, dept. of Electrical Engineering, MIT.

Wexler, J. D. Information networks in generative computer-assisted instruction, IEEE Transactions on Man-Machine Systems, 1970, MMS-11, 181-189.

Wortman, P. M. Medical diagnosis: an information-processing approach, Computers and Biomedical Research, 1972, 5, 315-328.

Ziegler, C. A. Individualized, monitored, syntax-directed computer-aided instruction in college and high-school algebra. Unpublished doctoral dissertation, 1972, University of California, Berkeley.

FILMED FROM BEST AVAILABLE COPY

(Continued from inside front cover)

96 R. C. Atkinson, J. W. Brelsford, and R. M. Shiffrin. Multi-process models for memory with applications to a continuous presentation task. April 13, 1966. (J. math. Psychol., 1967, 4, 277-300).

97 P. Suppes and E. Crothers. Some remarks on stimulus-response theories of language learning. June 12, 1966.

98 R. Bjork. All-or-none subprocesses in the learning of complex sequences. (J. math. Psychol., 1968, 1, 182-195).

99 E. Gammon. The statistical determination of linguistic units. July 1, 1966.

100 P. Suppes, L. Hyman, and M. Jerman. Linear structural models for response and latency performance in arithmetic. (In J. P. Hill (ed.), Minnesota Symposia on Child Psychology. Minneapolis, Minn.: 1967. Pp. 160-200).

101 J. L. Young. Effects of intervals between reinforcements and test trials in paired-associate learning. August 1, 1966.

102 H. A. Wilson. An investigation of linguistic unit size in memory processes. August 3, 1966.

103 J. T. Townsend. Choice behavior in a cued-recognition task. August 8, 1966.

104 W. H. Batchelder. A mathematical analysis of multi-level verbal learning. August 9, 1966.

105 H. A. Taylor. The observing response in a cued psychophysical task. August 10, 1966.

106 R. A. Bjork. Learning and short-term retention of paired associates in relation to specific sequences of interpresentation intervals. August 11, 1966.

107 R. C. Atkinson and R. M. Shiffrin. Some Two-process models for memory. September 30, 1966.

108 P. Suppes and C. Ihrke. Accelerated program in elementary-school mathematics--the third year. January 30, 1967.

109 P. Suppes and I. Rosenthal-Hill. Concept formation by kindergarten children in a card-sorting task. February 27, 1967.

110 R. C. Atkinson and R. M. Shiffrin. Human memory: a proposed system and its control processes. March 21, 1967.

111 Theodore S. Rodgers. Linguistic considerations in the design of the Stanford computer-based curriculum in initial reading. June 1, 1967.

112 Jack M. Knutson. Spelling drills using a computer-assisted instructional system. June 30, 1967.

113 R. C. Atkinson. Instruction in initial reading under computer control: the Stanford Project. July 14, 1967.

114 J. W. Brelsford, Jr. and R. C. Atkinson. Recall of paired-associates as a function of overt and covert rehearsal procedures. July 21, 1967.

115 J. H. Stelzer. Some results concerning subjective probability structures with semiorders. August 1, 1967

116 D. E. Rumelhart. The effects of interpresentation intervals on performance in a continuous paired-associate task. August 11, 1967.

117 E. J. Fishman, L. Keller, and R. E. Atkinson. Massed vs. distributed practice in computerized spelling drills. August 18, 1967.

118 G. J. Groen. An investigation of some counting algorithms for simple addition problems. August 21, 1967.

119 H. A. Wilson and R. C. Atkinson. Computer-based instruction in initial reading: a progress report on the Stanford Project. August 25, 1967.

120 F. S. Roberts and P. Suppes. Some problems in the geometry of visual perception. August 31, 1967. (Synthese, 1967, 17, 173-201)

121 D. Jamison. Bayesian decisions under total and partial ignorance. D. Jamison and J. Kozielecki. Subjective probabilities under total uncertainty. September 4, 1967.

122 R. C. Atkinson. Computerized instruction and the learning process. September 15, 1967.

123 W. K. Estes. Outline of a theory of punishment. October 1, 1967.

124 T. S. Rodgers. Measuring vocabulary difficulty: An analysis of item variables in learning Russian-English and Japanese-English vocabulary parts. December 18, 1967.

125 W. K. Estes. Reinforcement in human learning. December 20, 1967.

126 G. L. Wolford, D. L. Wessel, W. K. Estes. Further evidence concerning scanning and sampling assumptions of visual detection models. January 31, 1968.

127 R. C. Atkinson and R. M. Shiffrin. Some speculations on storage and retrieval processes in long-term memory. February 2, 1968.

128 John Holmgren. Visual detection with imperfect recognition. March 29, 1968.

129 Lucille B. Miodnosky. The Frostig and the Bander Gestalt as predictors of reading achievement. April 12, 1968.

130 P. Suppes. Some theoretical models for mathematics learning. April 15, 1968. (Journal of Research and Development in Education, 1967, 1, 5-22)

131 G. M. Olson. Learning and retention in a continuous recognition task. May 15, 1968.

132 Ruth Norene Hartley. An investigation of list types and cues to facilitate initial reading vocabulary acquisition. May 29, 1968.

133 P. Suppes. Stimulus-response theory of finite automata. June 19, 1968.

134 N. Moler and P. Suppes. Quantifier-free axioms for constructive plane geometry. June 20, 1968. (In J. C. H. Gerretsen and F. Oort (Eds.), Compositio Mathematica. Vol. 20. Groningen, The Netherlands: Wolters-Noordhoff, 1968. Pp. 143-152.)

135 W. K. Estes and D. P. Horst. Latency as a function of number or response alternatives in paired-associate learning. July 1, 1968.

136 M. Schlag-Rey and P. Suppes. High-order dimensions in concept identification. July 2, 1968. (Psychom. Sci., 1968, 11, 141-142)

137 R. M. Shiffrin. Search and retrieval processes in long-term memory. August 15, 1968.

138 R. D. Freund, G. R. Loftus, and R.C. Atkinson. Applications of multiprocess models for memory to continuous recognition tasks. December 18, 1968.

139 R. C. Atkinson. Information delay in human learning. December 18, 1968.

140 R. C. Atkinson, J. E. Holmgren, and J. F. Juola. Processing time as influenced by the number of elements in the visual display. March 14, 1969.

141 P. Suppes, E. F. Loftus, and M. Jerman. Problem-solving on a computer-based teletype. March 25, 1969.

142 P. Suppes and Mona Morningstar. Evaluation of three computer-assisted instruction programs. May 2, 1969.

143 P. Suppes. On the problems of using mathematics in the development of the social sciences. May 12, 1969.

144 Z. Domotor. Probabilistic relational structures and their applications. May 14, 1969.

145 R. C. Atkinson and T. D. Wickens. Human memory and the concept of reinforcement. May 20, 1969.

146 R. J. Titiev. Some model-theoretic results in measurement theory. May 22, 1969.

147 P. Suppes. Measurement: Problems of theory and application. June 12, 1969.

148 P. Suppes and C. Ihrke. Accelerated program in elementary-school mathematics—the fourth year. August 7, 1969.

149 D. Rundus and R.C. Atkinson. Rehearsal in free recall: A procedure for direct observation. August 12, 1969.

150 P. Suppes and S. Feldman. Young children's comprehension of logical connectives. October 15, 1969.

(Continued on back cover)

FILMED FROM BEST AVAILABLE COPY

( Continued from inside back cover )

151 Joaquim H. Laubsch. An adaptive teaching system for optimal item allocation. November 14, 1969.
152 Roberta L. Klatzky and Richard C. Atkinson. Memory scans based on alternative test stimulus representations. November 25, 1969.
153 John E. Holmgren. Response latency as an indicant of information processing in visual search tasks. March 16, 1970.
154 Patrick Suppes. Probabilistic grammars for natural languages. May 15, 1970.
155 E. Gammon. A syntactical analysis of some first-grade readers. June 22, 1970.
156 Kenneth N. Wexler. An automaton analysis of the learning of a miniature system of Japanese. July 24, 1970.
157 R. C. Atkinson and J.A. Paulson. An approach to the psychology of instruction. August 14, 1970.
158 R.C. Atkinson, J.D. Fletcher, H.C. Chetin, and C.M. Stauffer. Instruction in initial reading under computer control: the Stanford project. August 13, 1970.
159 Dewey J. Rundus. An analysis of rehearsal processes in free recall. August 21, 1970.
160 R.L. Klatzky, J.F. Juola, and R.C. Atkinson. Test stimulus representation and experimental context effects in memory scanning.
161 William A. Rottmayer. A formal theory of perception. November 13, 1970.
162 Elizabeth Jane Fishman Loftus. An analysis of the structural variables that determine problem-solving difficulty on a computer-based teletype. December 18, 1970.
163 Joseph A. Van Campen. Towards the automatic generation of programmed foreign-language instructional materials. January 11, 1971.
164 Jamesine Friend and R.C. Atkinson. Computer-assisted instruction in programming: AID. January 25, 1971.
165 Lawrence James Hubert. A formal model for the perceptual processing of geometric configurations. February 19, 1971.
166 J. F. Juola, I.S. Fischler, C.T. Wood, and R.C. Atkinson. Recognition time for information stored in long-term memory.
167 R.L. Klatzky and R.C. Atkinson. Specialization of the cerebral hemispheres in scanning for information in short-term memory.
168 J.D. Fletcher and R.C. Atkinson. An evaluation of the Stanford CAI program in initial reading ( grades K through 3 ). March 12, 1971.
169 James F. Juola and R.C. Atkinson. Memory scanning for words versus categories.
170 Ira S. Fischler and James F. Juola. Effects of repeated tests on recognition time for information in long-term memory.
171 Patrick Suppes. Semantics of context-free fragments of natural languages. March 30, 1971.
172 Jamesine Friend. Instruct coders' manual. May 1, 1971.
173 R.C. Atkinson and R.M. Shiffrin. The control processes of short-term memory. April 19, 1971.
174 Patrick Suppes. Computer-assisted instruction at Stanford. May 19, 1971.
175 D. Jamison, J.D. Fletcher, P. Suppes, and R.C. Atkinson. Cost and performance of computer-assisted instruction for compensatory education.
176 Joseph Offir. Some mathematical models of individual differences in learning and performance. June 28, 1971.
177 Richard C. Atkinson and James F. Juola. Factors influencing speed and accuracy of word recognition. August 12, 1971.
178 P. Suppes, A. Goldberg, G. Kanz, B. Searle, and C. Stauffer. Teacher's handbook for CAI courses. September 1, 1971.
179 Adele Goldberg. A generalized instructional system for elementary mathematical logic. October 11, 1971.
180 Max Jerman. Instruction in problem solving and an analysis of structural variables that contribute to problem-solving difficulty. November 12, 1971.
181 Patrick Suppes. On the grammar and model-theoretic semantics of children's noun phrases. November 29, 1971.
182 Georg Kreisel. Five notes on the application of proof theory to computer science. December 10, 1971.
183 James Michael Moloney. An investigation of college student performance on a logic curriculum in a computer-assisted instruction setting. January 28, 1972.
184 J.E. Friend, J.D. Fletcher, and R.C. Atkinson. Student performance in computer-assisted instruction in programming. May 10, 1972.
185 Robert Lawrence Smith, Jr. The syntax and semantics of ERICA. June 14, 1972.
186 Adele Goldberg and Patrick Suppes. A computer-assisted instruction program for exercises on finding axioms. June 23, 1972.
187 Richard C. Atkinson. Ingredients for a theory of instruction. June 26, 1972.
188 John D. Bonvillian and Veda R. Charrow. Psycholinguistic implications of deafness: A review. July 14, 1972.
189 Phipps Arabie and Scott A. Boorman. Multidimensional scaling of measures of distance between partitions. July 26, 1972.
190 John Ball and Dean Jamison. Computer-assisted instruction for dispersed populations: System cost models. September 15, 1972.
191 W. R. Sanders and J. R. Ball. Logic documentation standard for the Institute for Mathematical Studies in the Social Sciences. October 4, 1972.
192 M.T. Kane. Variability in the proof behavior of college students in a CAI course in logic as a function of problem characteristics. October 6, 1972.
193 P. Suppes. Facts and fantasies of education. October 18, 1972.
194 R. C. Atkinson and J. F. Juola. Search and decision processes in recognition memory. October 27, 1972.
195 P. Suppes, R. Smith , and M. Léveillé. The French syntax and semantics of PHILIPPE, part 1 : Noun phrases. November 3, 1972.
196 D. Jamison, P. Suppes, and S. Wells. The effectiveness of alternative instructional methods: A survey. November 1972.
197 P. Suppes. A survey of cognition in handicapped children. December 29, 1972.
198 B. Searle, P. Lorton, Jr., A. Goldberg, P. Suppes, N. Ledet, and C. Jones. Computer-assisted instruction program: Tennessee State University. February 14, 1973.
199 D. R. Levine. Computer-based analytic grading for German grammar instruction. March 16, 1973.
200 P. Suppes, J.D. Fletcher, M. Zanotti, P. V. Lorton, Jr., and B. W. Searle. Evaluation of computer-assisted instruction in elementary mathematics for hearing-impaired students. March 17, 1973.
201 G. A. Huff. Geometry and formal linguistics. April 27, 1973.
202 C. Jensema. Useful techniques for applying latent trait mental-test theory. May 9, 1973.
203 A. Goldberg. Computer-assisted instruction: The application of theorem-proving to adaptive response analysis. May 25, 1973.
204 R. C. Atkinson, D. J. Herrmann, and K. T. Wescourt. Search processes in recognition memory. June 8, 1973.
205 J. Van Campen. A computer-based introduction to the morphology of Old Church Slavonic. June 18, 1973.
206 R. B. Kimball. Self-optimizing computer-assisted tutoring: Theory and practice. June 25, 1973.